GUIDA ALL' USO DEL SISTEMA

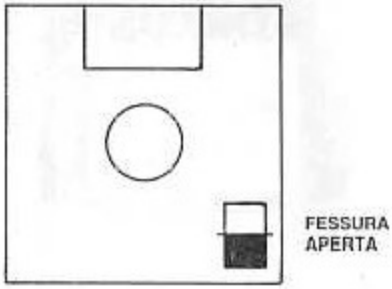
PC 128 S

olivetti PRODEST

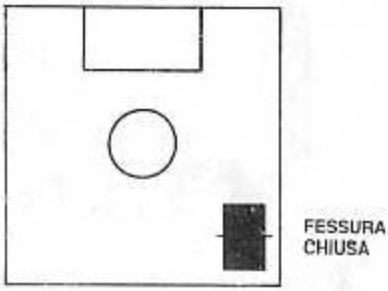
PC 128 S AVVERTENZE IMPORTANTI!!!

LEGGERE PRIMA DELL'USO

1) Il disco è protetto in scrittura quando la linguetta di protezione è posizionata verso il basso (o esterno), cioè quando la fessura rettangolare è aperta.



Protetto in scrittura



Non-protetto in scrittura

È un'ottima abitudine tenere il più possibile la linguetta nella posizione "Protetto". Ciò permette di evitare che il PC 128 S scriva o cancelli accidentalmente delle informazioni su disco.

Si consiglia, inoltre, ogni qualvolta si copiano delle informazioni, di apporre la protezione in scrittura sul disco che verrà copiato per evitare che il disco venga accidentalmente alterato.

2) Dopo aver installato ed acceso il vostro PC 128 S (Vedi Cap. 1 e 2) eseguire subito una copia del disco Welcome procedendo come segue:

   a) Inserire il disco Welcome nel drive assicurandosi che sia protetto in scrittura. Si ricorda che prima di effettuare ogni operazione è necessario attendere che la luce del drive sia spenta.

   b) Caricare il disco Welcome premendo SHIFT + BREAK rilasciando per primo il tasto BREAK.

   c) Selezionare "ADFS" dal menu "Utilità".

   d) Attivare "S.D. Backup" e premere un tasto qualsiasi (mai BREAK).

   e) Premere di nuovo un tasto e all'apparire di "please insert DESTINATION disc and press a key" togliere il disco Welcome (SOURCE) e inserire il disco su cui fare la copia (DESTINATION) assicurandosi che non sia protetto.

   f) Premere un tasto e quando il monitor visualizza "please insert SOURCE disc and press a key" effettuare di nuovo il cambio dei dischi.

   g) Procedere come in "e" ed "f" finché non appaia sul monitor "Backup. completed - Please remove the disc and then press break".
   A questo punto togliere il disco e premere BREAK.

L'operazione è così terminata.

MONITOR
AVVERTENZE IMPORTANTI!!!

1) **IMMAGINE SPOSTATA VERSO L'ALTO O VERSO IL BASSO:**

Digitare:

a) *CO. TVn,1

dove n è: 1 - due righe in su
0 - una riga in su
255 - normale

b) CTRL + BREAK

In alternativa agire, per mezzo di un cacciavite, sul controllo *vertical height* situato sul retro dell'unità.

2) **IMMAGINE LEGGERMENTE INCLINATA**

Questo fenomeno è generalmente dovuto all'influenza del magnetismo terreste. Ruotare l'orientamento del monitor finché l'immagine non torni normale.

3) **IMMAGINE NON FERMA**

Con l'uso di un cacciavite agire sul controllo *vertical hold*, situato sul retro dell'unità, fino a stabilizzare l'immagine.

olivetti PRODEST

PC 128 S

GUIDA ALL' USO DEL SISTEMA

DC 5040
Versione 1.0
Settembre 1986

Prodotto da Olivetti Prodest S.p.A.

Tutti i diritti riservati. Nessuna parte del manuale e dei programmi può essere duplicata, copiata, trasmessa o riprodotta in qualsiasi forma o con qualsiasi mezzo senza il preventivo consenso scritto della Olivetti Prodest S.p.A.

Olivetti Prodest S.p.A.
Via Caldera, 21 - 20153 Milano (MI)
Telefono 02/452731

© 1986 Acorn Computers Limited

Importante:

I fili per il collegamento elettrico dell'unità base del personal computer Olivetti Prodest PC 128 S sono colorati secondo il seguente codice:

Verde e giallo	**Terra**
Blu	**Neutro**
Marrone	**Tensione**

Se la presa di cui si dispone non è adatta alla spina in dotazione, quest'ultima deve essere sostituita con una spina appropriata collegata secondo i codici sopra indicati, o si deve ricorrere ad apposito adattatore disponibile presso qualsiasi negozio di apparecchiature elettriche.

Ambiente operativo

Il personal computer non deve essere esposto alla luce diretta del sole o all'umidità per periodi di tempo prolungati.

Aerazione

Non bloccare le fessure di ventilazione della carrozzeria. Per dettagli al riguardo, consultare il testo.

All'interno di questa pubblicazione viene usato il termine BBC come abbreviazione di British Broadcasting Corporation.

Le informazioni contenute in questo manuale, o il prodotto in esso descritto, non possono essere nè per intero nè in parte adattate o riprodotte in qualsiasi forma, senza previa approvazione scritta della Olivetti Prodest.

Il personal computer Olivetti Prodest PC 128 S descritto in questo manuale e i prodotti direttamente connessi al suo funzionamento sono soggetti a continui sviluppi e miglioramenti. Tutte le informazioni di natura tecnica, i particolari sul prodotto e il suo uso (comprese le informazioni e i particolari inclusi in questo manuale) sono forniti dalla Olivetti Prodest con la certezza di aver operato per il meglio. Purtuttavia, si riconosce la possibilità che in questo manuale ci possano essere degli errori o delle omissioni. Una lista dettagliata di ogni correzione o revisione operata su questo manuale si può ottenere su richiesta dalla Olivetti Prodest. La Olivetti Prodest accoglierà con favore qualsiasi commento o suggerimento riguardante il prodotto e il manuale ad esso relativo.

La corrispondenza va indirizzata a:

Olivetti Prodest
Via Caldera, 21
20153 MILANO

La manutenzione e l'assistenza del prodotto in questione sono a carico dei rivenditori autorizzati o dei centri di assistenza Olivetti. La Olivetti Prodest non accetta alcuna responsabilità per qualsiasi perdita o danno causati da personale non autorizzato. Lo scopo di questo manuale è unicamente quello di assistere il lettore nell'uso del PC 128 S, pertanto la Olivetti Prodest non si assume alcuna responsabilità per qualsiasi perdita o danno derivanti da un uso non corretto del prodotto oppure dall'uso di informazioni, particolari, oppure di eventuali errori od omissioni, contenuti nel presente manuale.

Olivetti Prodest è un marchio di fabbrica della Olivetti Prodest S.p.A.
VIEW e ViewSheet sono marchi di fabbrica della Acornsoft Limited

Indice

Prefazione 1

Introduzione 3

1. Come iniziare 5

Uso del computer 8
Come comunicare con il computer 15
Uso del disk drive 17

2. Utilizzo dei programmi Welcome ad icone 19

Caricamento dei programmi Welcome 19
Funzionamento del menu Welcome 19
Altri programmi Welcome 25
Grafica 26
Tutorials 27
Applicazioni 27
Giochi 31
Programmi di utilità 31
Dispositivo/colore 37

3. Il linguaggio BASIC 39

Come scrivere un programma 39
Un semplice programma con l'utilizzo di variabili 41
Supporto del BBC BASIC alla programmazione 45
Memorizzazione e caricamento di un programma 49
Programmare o non programmare 50
Grafica 51
Stampa del testo 58
Input 65
Programmi strutturali 67
Funzioni 73
Loop 74
Scelte 78
Gestione errori 82
Ulteriori informazioni sulle stringhe 83
Matrici 88
File 90
Ulteriori informazioni sulle funzioni grafiche 92
Modalità teletext 97
Suono 100
BASIC 128K 102
Linguaggio Assembler 102

4. Introduzione a VIEW 105

Che cos'è un word processor 105
Il word processor VIEW 105
Stampa da VIEW 124
Funzioni supplementari di VIEW 126

5. Introduzione a ViewSheet 129

Che cos'è uno spreadsheet 129
ViewSheet 131
Uso di ViewSheet 131
Stampa 148
Altre funzioni di ViewSheet 149

6. Sistemi di archiviazione 151

Che cos'è un sistema di archiviazione 151
Sistemi di archiviazione standard 151
Sistema di archiviazione ROM (RFS) 152
Sistema di archiviazione avanzato su disco (ADFS) 152
Uso di ADFS 157
Directory 160
Ulteriori informazioni sulle directory 163
Comandi del sistema operativo MOS riguardanti i sistemi di archiviazione 169

7. Espansione del sistema 175

Collegamento di un monitor a colori e di un monitor monocromatico 175
Collegamento di un televisore 175
Disk drive 176
Collegamento di una stampante 176
L'interfaccia RS232 177
Collegamento di un joystick, mouse, trackball 178
La rete Econet 178

Appendici

1 Informazioni tecniche 179
2 Caratteristiche dei modi 182
3 Set di caratteri 184
4 Comandi del sistema operativo 192
5 Comandi *FX 197
6 Codici VDU 205
7 Comandi del sistema di archiviazione 210
8 Parole chiave del BASIC 215
9 Codici PLOT 223
10 Informazioni sul BAS128 225
11 Comandi VIEW 227
12 Comandi di ViewSheet 231
13 Albero del disco Welcome 233
14 Caratteri accentati 235

Prefazione

Fino ad alcuni anni fa, l'idea di avere un computer in casa sarebbe stata accolta con incredulità. Ai nostri giorni, i personal computer sono un dato di fatto e un numero sempre crescente di persone desidera saperne di più sulle loro reali possibilità. Fortunatamente gli anni trascorsi hanno visto molti cambiamenti e i microcomputer dell'attuale generazione offrono una reale capacità di elaborazione, che li rende adatti ad un uso non solo hobbistico, ma anche scolastico e tecnico–scientifico, senza contare l'utilizzo in uffici che con sempre maggiore frequenza adottano l'automazione.

Anche i sistemi di microcomputer Olivetti sono stati soggetti a questo lungo periodo di cambiamenti e, in conseguenza di un processo di continuo sviluppo, abbracciano attualmente un'ampia gamma di funzioni sofisticate. Il personal computer appena acquistato è già provvisto di alcuni dei programmi più sofisticati previsti per il microcomputer Olivetti Prodest PC 128 S, ma va ricordato che esistono migliaia di altri programmi che potenzialmente potranno essere usati con questo computer.

Benvenuti nel mondo del personal computer Olivetti Prodest PC 128 S.

Introduzione

Questo manuale costituisce un'introduzione al personal computer Olivetti Prodest PC 128 S e fornisce le informazioni fondamentali a tutti i nuovi utenti, compresi coloro che non hanno mai usato un computer.

Il manuale comprende l'impostazione iniziale del sistema, le informazioni su come fare sì che il computer esegua i comandi impartiti, i dettagli dei programmi forniti insieme al sistema e appendici riguardanti funzioni specifiche del computer. Questa pubblicazione non intende essere un manuale tecnico di consultazione o una guida di auto–apprendimento. Gli utenti che desiderino questo tipo di materiale dovranno rivolgersi al proprio fornitore o alla Olivetti Prodest per informazioni sulle pubblicazioni esistenti.

Il primo capitolo descrive come iniziare ad usare il PC 128 S e illustra la disposizione e l'uso della tastiera. I capitoli seguenti introdurranno:

- i programmi "Welcome" fra cui il pacchetto di elaborazione testi VIEW ed il pacchetto di foglio elettronico ViewSheet
- il linguaggio BASIC BBC
- il sistema di archiviazione su disco ADFS
- le possibilità di espansione del computer.

1. Come iniziare

Come disimballare il computer

Il PC 128 S viene fornito in tre scatole distinte, che, oltre al presente manuale, contengono:

Una scatola:

– **una unità di visualizzazione** simile a un televisore (monitor)
– il tagliando di garanzia
– il libretto di istruzioni

L'altra scatola:

– **l'unità centrale completa di tastiera**
– il disco "Welcome"
– il tagliando di garanzia
– la scheda di riferimento VIEW
– la scheda di riferimento ViewSheet
– la scheda tasti funzione VIEW/ViewSheet

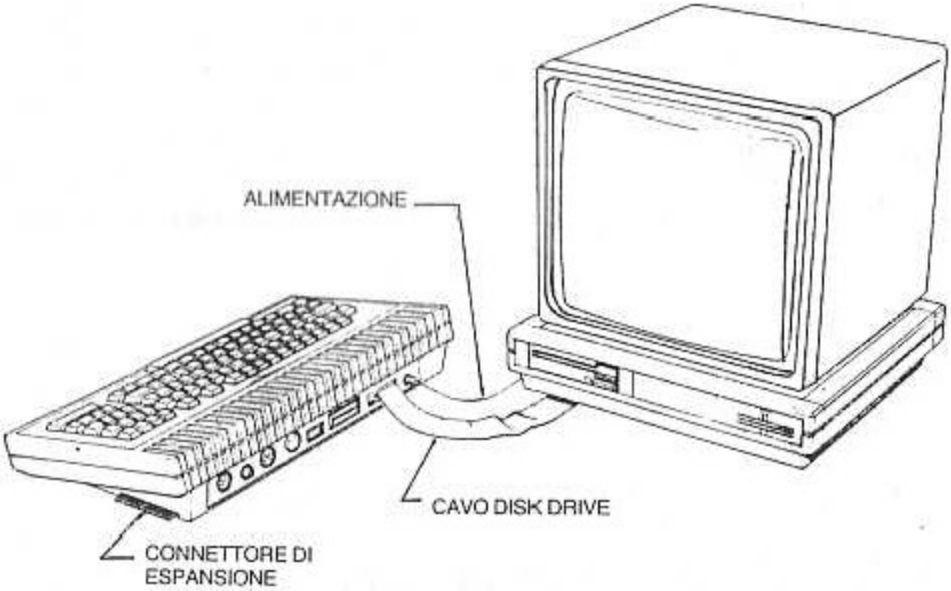
La terza scatola:

– **l'unità con il disco** (drive)

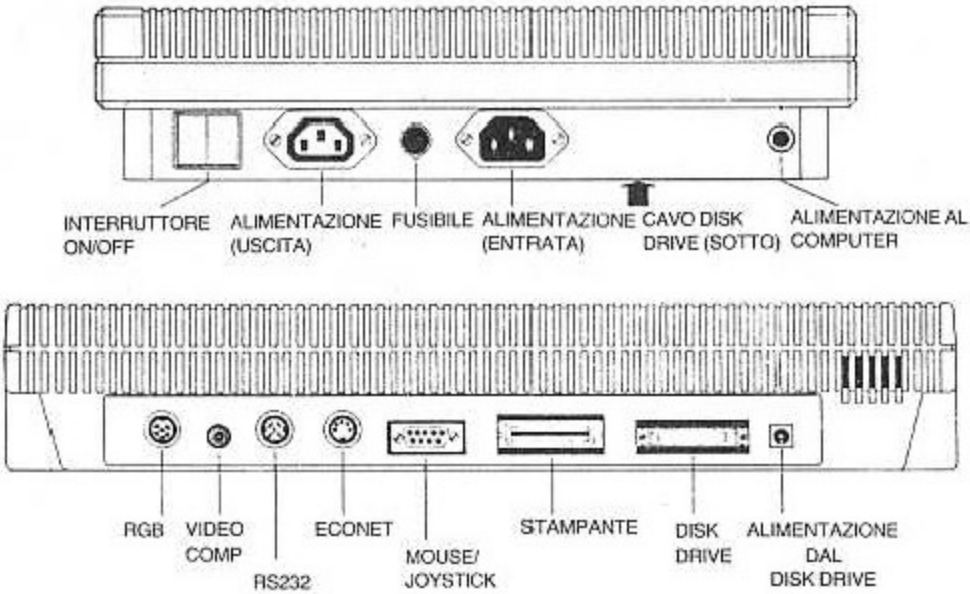
Compilare i tagliandi di garanzia seguendo le avvertenze indicate. Disimballare tutti i pezzi sopra elencati, quindi appiattire le scatole di cartone e conservarle insieme agli elementi di polistirolo in modo che, nel caso improbabile di un guasto, il computer possa essere restituito senza inconvenienti al fornitore. Se dovesse mancare qualcuno degli elementi sopra elencati, contattare immediatamente il fornitore.

Collegamento dei componenti

Collocare l'unità centrale di fronte al drive su una superficie orizzontale piatta. Evitare superfici morbide, come moquette, che potrebbero bloccare le fessure di ventilazione della carrozzeria. Dovranno esserci due prese di corrente nelle vicinanze, e un certo spazio fra la parete e la parte posteriore del sistema, per consentire un agevole accesso ai cavi.



Seguendo la figura riportata di seguito, collegare i cavi dell'alimentazione e dell'unità drive alle corrispondenti prese dell'unità centrale, facendo passare il cavo dal retro al di sotto del computer stesso. Collegare direttamente alla presa di corrente l'unità drive. A seconda della presa, collegare l'unità di visualizzazione direttamente alla presa di corrente di rete o al connettore posto nella parte posteriore dell'unità drive del PC 128 S.

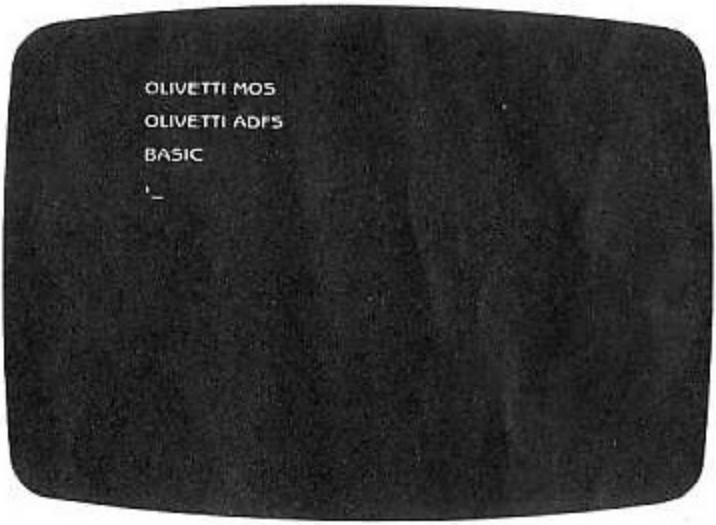


L'unità di visualizzazione dovrà essere collegata o alla presa "RGB" o a quella "Comp Video", collocate nella parte posteriore dell'unità centrale: generalmente gli schermi a colori vanno collegati alla presa "RGB" e quelli monocromatici alla presa "Comp Video", ma in caso di incertezza consultare le istruzioni che accompagnano l'unità di visualizzazione. Prima di accendere il computer, occorre regolare i controlli di luminosità e il contrasto sull'unità di visualizzazione in una posizione intermedia, se necessario servendosi di nuovo della documentazione allegata.

Accensione

A questo punto è possibile accendere il computer con l'interruttore ON/OFF posto nella parte posteriore dell'unità a disco (consultare la figura precedente). Quando si preme l'interruttore si dovrà sentire un suono acuto proveniente dall'altoparlante del computer e le due spie luminose rosse dovranno accendersi sulla parte anteriore della tastiera.

Subito dopo, sullo schermo dovrà apparire il seguente messaggio:



Il computer è ora pronto per l'uso, e si può passare alla lettura della sezione intitolata "*Uso del computer*". Se sullo schermo non appare il messaggio in questione, consultare la seguente guida all'individuazione di errori.

Individuazione di errori

La prima cosa da controllare è che l'unità drive e lo schermo siano collegati alle prese di corrente e accesi. Una volta accertato ciò, assicurarsi che sia correttamente stabilito il collegamento elettrico dall'unità drive alla unità centrale.

Consultare le istruzioni fornite insieme al video per accertarsi che quest'ultimo sia collegato correttamente alla rete o al connettore posto sulla parte posteriore dell'unità drive del PC 128 S. Controllare che i controlli di Luminosità e/o di Contrasto non siano posti al livello minimo, cosa che impedirebbe di vedere qualsiasi immagine generata dal computer.

A questo punto le due spie luminose rosse sulla tastiera dovranno essere accese e sullo schermo dovrebbe apparire l'immagine riprodotta nella figura precedente. In caso contrario, contattare il proprio fornitore o la Olivetti Prodest per una consulenza o per l'assistenza assicurata dalla garanzia.

USO DEL COMPUTER

Questa sezione è stata concepita per consentire una ulteriore familiarizzazione con il funzionamento del personal computer a livello elementare, vale a dire usando la tastiera per impartire comandi semplici. Verrà descritta inizialmente la tastiera per poi passare alle convenzioni adottate per descrivere, nelle sezioni successive, l'utilizzo di determinati tasti.

La tastiera

Per facilitarne la descrizione, la tastiera del computer può essere divisa in quattro aree separate:

- la tastiera principale **alfa–numerica** che ha una disposizione dei tasti analoga a quella delle normali macchine per scrivere, con qualche tasto in più
- un **tastierino numerico** che contiene tasti associati con l'introduzione di dati numerici
- un gruppo di tasti di **controllo cursore**
- una fila di **tasti funzione** contrassegnati da f0 a f9.

Il tocco richiesto da questa tastiera è simile a quello necessario a gran parte delle macchine per scrivere elettriche, in quanto basta una breve, leggera pressione per attivare ciascun tasto. La differenza, ovviamente, consiste nel fatto che i caratteri prodotti da ogni pressione di tasto sono visualizzati sullo schermo e non vengono stampati su carta. In circostanze normali, la risposta è immediata, sebbene ci siano occasioni (quando per esempio il computer sta già eseguendo un'altra operazione) in cui può trascorrere un breve lasso di tempo prima che i caratteri appaiano sullo schermo.

La tastiera include anche una funzione di **ripetizione automatica.** Se si tiene premuto un tasto, il carattere corrispondente verrà ripetuto sullo schermo, dopo un breve lasso di tempo iniziale. La ripetizione del carattere continua finchè non si rilascia il tasto o finchè il computer esaurisce lo spazio disponibile per memorizzare la riga introdotta (questa condizione viene segnalata dall'emissione di un suono continuo dall'altoparlante).

D'ora in poi verrà usata

questa grafia

per indicare un input dalla tastiera e, conseguentemente, un output sullo schermo, e simboli come

RETURN

per indicare la pressione di tasti specifici. La pressione simultanea di due tasti è indicata nel modo seguente:

SHIFT + BREAK

La tastiera alfa–numerica

La tastiera alfa–numerica contiene tasti indicanti tutte le lettere dell'alfabeto (compreso lo spazio), i numeri da 0 a 9, vari simboli speciali (come la punteggiatura, %, ecc.) più un certo numero di altri tasti con funzioni particolari. Contiene anche, nell'angolo in basso a sinistra, una fila di tre spie rosse luminose contrassegnate *on, caps lock* e *shift lock.*

La spia dell'alimentazione (*on*) si illumina quando il computer è acceso.

Se *caps lock* è attivato (vale a dire, se la spia corrispondente è accesa), la pressione di qualsiasi tasto alfabetico produrrà una lettera maiuscola; la pressione di qualsiasi tasto contenente due simboli produrrà il carattere inferiore.

Se *shift lock* è attivato, i tasti alfabetici produrranno ancora lettere maiuscole, ma la pressione di qualsiasi tasto contenente due simboli produrrà il carattere superiore.

Se nessuna di queste due funzioni è attivata, la pressione dei tasti alfabetici produrrà lettere minuscole e i tasti contenenti due simboli produrranno il carattere inferiore.

Lo stato delle spie luminose *caps lock* e *shift lock* è controllato dai tasti

CAPS LOCK e SHIFT LOCK

Ciascuna pressione provoca l'accensione o lo spegnimento della corrispondente spia luminosa, a seconda del suo stato corrente. Notare che è impossibile provocare dalla tastiera contemporaneamente l'accensione di caps lock e di shift lock. Il PC 128 S si serve di questa indicazione simultanea per evidenziare una particolare situazione, descritta più avanti.

I due tasti SHIFT non hanno alcun effetto quando è attivato *shift lock*. Se *shift lock* è disattivato (indipendentemente dalla impostazione di *caps lock*), i tasti SHIFT producono lettere maiuscole e simboli se uno dei due viene premuto contemporaneamente a un altro tasto. I tasti SHIFT non influenzano la spia *shift lock*.

Un'ulteriore opzione viene fornita premendo SHIFT + CAPS LOCK

In questo caso, caps lock viene attivato nel modo usuale, ma durante l'introduzione di caratteri successivi si potranno ottenere lettere minuscole tenendo premuto un tasto SHIFT.

Il fatto che *caps lock* o *shift lock* (o nessuno dei due tasti) sia attivato per una particolare sessione al computer, è una questione di scelta personale, sebbene la decisione dipenda anche dal tipo di input, ad esempio:

- Un testo convenzionale, come ad esempio una lettera introdotta nell'elaboratore testi VIEW, consiste principalmente di caratteri minuscoli combinati con poche lettere maiuscole.

- Un programma BASIC consiste di un insieme di speciali parole maiuscole (chiamate parole chiave) combinate con altre parole, spesso minuscole (chiamate nomi di variabile).

CTRL (che è un'abbreviazione di CONTROL) non ha effetto se usato singolarmente, ma può essere utilizzato in combinazione con altri tasti per produrre un certo numero di effetti speciali. Per esempio, CTRL + G fa in modo che il computer emetta un breve suono; CTRL + L cancella ciò che compare sullo schermo. Più avanti, in questo manuale, vengono forniti altri esempi e nell'Appendice 6 viene dato un sommario dei vari effetti possibili.

: permette di generare dei caratteri non previsti sulla tastiera e viene utilizzato soprattutto nella scrittura in lingua straniera. Se premuto contemporaneamente a SHIFT e CTRL il carattere battuto successivamente verrà interpretato dal computer come simbolo speciale.

TAB generalmente ha lo stesso effetto della barra spaziatrice, sebbene acquisti una funzione particolare se si utilizza il programma di elaborazione testi VIEW, come descritto più avanti nel presente manuale.

RETURN viene usato per indicare che una particolare riga di input è completa. Prima di premere RETURN, DELETE può essere usato per cancellare il carattere, o i caratteri, che si sono appena battuti.

ESCAPE e BREAK, come indica la loro denominazione, vengono usati per consentire un'interruzione dell'attività del computer, sebbene ESCAPE assuma una funzione particolare quando si usa VIEW. ESCAPE va considerato un tasto dall'effetto piuttosto blando, che generalmente arresta l'attività del computer senza effetti collaterali di rilievo, mentre BREAK comporta un drastico arresto del computer stesso, indipendentemente dall'attività che sta eseguendo.

La pressione del solo BREAK viene spesso definita come un "**soft break**", poiché ha l'effetto di reinizializzare il computer e cioè di riportarlo alle condizioni in cui era prima della sessione corrente (BASIC, VIEW, ecc.). Un "**hard break**" viene attivato premendo CTRL + BREAK; ciò reimposta il computer in modo che assuma lo stato nel quale normalmente si troverebbe subito dopo l'accensione.

SHIFT + BREAK ha una funzione particolare. Ulteriori dettagli verranno forniti nella sezione intitolata "*Utilizzo dei programmi Welcome a icone*".

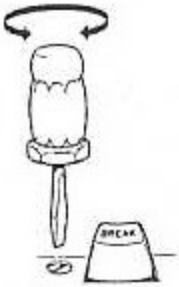
Un'altra importante funzione è costituita dalla pressione del tasto "R" contemporaneamente all'accensione ON/OFF del computer posta sul retro del drive. Ciò permette di riportare il computer allo stato originario quale all'atto dell'acquisto.

Sul monitor apparirà:

Options reset.
Press break to continue

Digitare BREAK per riniziare la seduta.

Gli eventuali effetti dannosi di una pressione accidentale di BREAK possono essere evitati girando di 90 gradi il *fermo del tasto di interruzione* (vedere l'illustrazione) in senso orario, servendosi di un cacciavite. Riportando la vite nella sua posizione originale viene ripristinato il funzionamento normale.



Il tastierino numerico

Il tastierino numerico costituisce un utile mezzo per introdurre grandi quantità di dati numerici. Contiene:

- Le cifre da 0 a 9;
- I simboli che contrassegnano le quattro operazioni aritmetiche (per la moltiplicazione viene usato *, per la divisione /);
- Il punto (punto decimale) e la virgola;
- Tasti RETURN e DELETE separati;
- Il simbolo #.

Ogni tasto riproduce la funzione del tasto corrispondente nella tastiera principale, con l'ulteriore vantaggio che i segni +, –, *, / e # possono essere ottenuti direttamente (vale a dire senza l'uso di SHIFT).

Tasti di controllo cursore

In circostanze normali, lo schermo mostra un simbolo lampeggiante, detto cursore, che indica la posizione nella quale apparirà il successivo carattere battuto. Il cursore si sposta verso destra di un carattere per ogni singola pressione, verso sinistra di un carattere per ogni pressione di DELETE e all'inizio di una nuova riga per ogni pressione di RETURN.

I quattro tasti freccia possono essere usati per spostare il cursore in qualsiasi punto dello schermo e nei capitoli successivi si noterà che questa funzione è fondamentale per l'uso di VIEW.

COPY ha un compito particolare in ognuna delle funzioni sopra descritte, ma viene anche usato insieme ai tasti di controllo cursore per effettuare modifiche principalmente durante l'introduzione e la correzione di programmi. (Vedasi il capitolo sul linguaggio BASIC).

I tasti funzione da f0 a f9

In alcune applicazioni, come ad esempio VIEW, è conveniente fare uso di una singola pressione di tasto per indicare una particolare azione e i 10 tasti funzione nella parte superiore della tastiera principale servono a questo scopo. Ogni tasto può essere usato singolarmente, insieme a SHIFT, CTRL o SHIFT + CTRL, fornendo in tal modo un totale di 40 funzioni addizionali di tastiera. In questi casi, è procedura usuale annotare la funzione attivata da ciascun tipo di pressione su un'apposita scheda di tastiera come quella fornita insieme al computer.

Inoltre, i tasti funzione possono essere programmati per produrre una sequenza di uno o più caratteri, consentendo in tal modo di ridurre al minimo il numero di pressioni di tasto necessarie per eseguire compiti da svolgere frequentemente. Una breve descrizione della programmazione di tasti funzione viene fornita nella sezione successiva e le istruzioni complete (incluso il modo in cui i tasti di controllo cursore COPY, e il tastierino numerico possono essere programmati) sono contenute nei manuali tecnici del PC 128 S.

Visualizzazione su schermo

Questa sezione introduce le varie visualizzazioni su schermo disponibili e fornisce la possibilità di esercitarsi all'uso della tastiera. Ad ogni modo evitare, per ora, di interessarsi al significato di ciò che si sta battendo; concentrarsi, piuttosto, sulla pressione dei tasti corretti. Se si batte una riga in modo errato (se per esempio si batte RETURN prima di individuare l'errore), il computer darà in risposta un semplice messaggio, come:

Mistake

o

No such variable

Ignorare per ora questi messaggi e limitarsi a battere di nuovo la riga; il significato dei messaggi in questione viene spiegato nei capitoli successivi. Uno degli errori più frequenti in questa fase è di battere la lettera O invece del numero 0, che sono contrassegnati rispettivamente da O e 0. Se si sono commessi errori di entità maggiore, provare a premere ESCAPE e, se ciò non ha alcun effetto, premere BREAK.

Il PC 128 S è in grado di visualizzare output sullo schermo in una varietà di diversi modi di visualizzazione, ognuno dei quali possiede caratteristiche proprie, per quanto riguarda il numero e la lunghezza delle righe di testo, le dimensioni e la forma dei caratteri visualizzati e la capacità di **visualizzazione grafica** (punti, linee e aree di colore). Ciascun modo di schermo è identificato da un numero, che può essere compreso in una gamma da 0 a 7 o da 128 a 135. Questi due insiemi di "modi" sono identici per quanto riguarda ciò che appare sullo schermo e differiscono solo per quanto concerne la dimensione e l'ubicazione dell'area di memoria utilizzata per memorizzare il contenuto corrente dello schermo. I modi da 0 a 7 sono conosciuti come i modi standard; i modi da 128 a 135 sono conosciuti come modi ombra o **shadow** e forniscono la capacità massima di memoria utente per un dato tipo di visualizzazione. Si prega di consultare l'Appendice 2 per una descrizione dei vari modi, sia standard che ombra. In quasi tutti gli esempi forniti in questa guida si utilizzeranno i modi ombra da 128 a 135.

Si dispone a questo punto di un mezzo che consente un utilizzo del computer in uno qualsiasi dei modi disponibili, ma l'impostazione standard è il modo 129, equivalente al modo 1 standard, che fornisce:

- 32 righe di testo, ciascuna delle quali ha una lunghezza di 40 caratteri;
- il set di caratteri **ASCII** (vedere più avanti);
- una grafica a quattro colori ad alta risoluzione.

Il simbolo > posto sulla sinistra del cursore lampeggiante è un esempio di **prompt** (messaggio guida) e la sua comparsa indica che il computer attende che si batta qualcosa. Provare a battere queste righe per vedere l'effetto; in ciascun caso il computer darà in risposta la visualizzazione dei caratteri fra virgolette:

PRINT"Bianco su nero" RETURN

Nel modo 135, SHIFT + f1 e CTRL + f2 ecc. generano ciò che è conosciuto come **codici di controllo teletext**, che influenzano il modo in cui vengono visualizzati i caratteri rimanenti su una riga particolare. Esempi di questo tipo di visualizzazione su schermo si possono trovare nella sezione di questo manuale dedicata al linguaggio BASIC. Battere per esempio:

MODE 135 RETURN
PRINT" SHIFT + f1 Rosso su nero" RETURN

Se si batte ora:

MODE128 RETURN

il contenuto dello schermo scomparirà e nell'angolo in alto a sinistra apparirà un prompt più piccolo.

A questo punto è stato selezionato il modo 128, che fornisce:

– 32 righe di testo, ciascuna con una lunghezza di 80 caratteri;
– l'intero set di caratteri ASCII (vedere più avanti);
– grafica a due colori, ad alta risoluzione.

A questo punto battere:

PRINT"Bianco su nero" RETURN
COLOUR0:COLOUR129:PRINT"Nero su bianco" RETURN
MOVE 600,500:PLOT149,750,500 RETURN
MOVE 600,500:PLOT157,700,500 RETURN

E' possibile ora provare a ripetere la stessa sequenza di esempi nei modi 130, 132 e 133, gli altri modi che forniscono una funzione grafica. Notare l'effetto che ciascun cambiamento di modo ha sulle dimensioni e sulla forma di ciascun carattere battuto, sui colori prodotti e sulla regolarità del tracciamento di un cerchio.

I modi 131 e 134 offrono esclusivamente una visualizzazione di testo, consistente in 25 righe, rispettivamente di 80 e 40 colonne.

Il programma Welcome contiene una dimostrazione delle capacità dei vari modi di schermo e l'Appendice 2 fornisce una descrizione completa delle caratteristiche di ciascun modo.

Avvertenza sui set di caratteri

I computer utilizzano semplici codici per rappresentare caratteri che sono immagazzinati in memoria o visualizzati sullo schermo. Il computer in questione offre due convenzioni di codice che hanno una diffusione internazionale, **teletext** e **ASCII** (ASCII è l'abbreviazione di American Standard Code for Information Interchange). Il set teletext è disponibile solo nei modi 7 e 135 e il set ASCII è disponibile in tutti gli altri.

È il set di caratteri ASCII che è rappresentato sui tasti del computer. Ogniqualvolta si prema un tasto, verrà visualizzata sullo schermo una rappresentazione del carattere corrispondente.

Per quanto riguarda il set di caratteri Teletext, questi è identico al set ASCII per tutte le lettere dell'alfabeto, per le cifre da 0 a 9 e per tutti i simboli speciali, tranne otto. Le differenze sono elencate sotto:

simbolo ASCII:	[	\	]	^	{	|	}	~
simbolo Teletext:	←	½	→	↑	¼	‖	¾	÷

Per notare la differenza dei simboli nei due modi premere i simboli ASCII dopo aver battuto:

MODE 135 RETURN

Verranno visualizzati i corrispondenti simboli Teletext.

Inoltre, il set di caratteri teletext contiene i caratteri con grafica elementare e alcuni codici di controllo teletext. Dettagli completi su questo argomento vengono forniti nell'Appendice 3.

Una ulteriore complicazione deriva dal fatto che il computer in questione consente una ridefinizione e un ampliamento del set di caratteri ASCII, permettendo in tal modo la visualizzazione di caratteri stranieri, corsivi e definiti dell'utente. L'esempio che segue consente la ridefinizione del tasto A in modo che visualizzi il simbolo matematico usato per indicare p greco:

MODE134 RETURN
VDU23,65,0,2,124,168,40,40,40,0 RETURN

Dopo aver osservato l'effetto p greco, premere CTRL + BREAK per riattivare la codifica nomale del carattere "A".

Una funzione che consente di progettare caratteri personalizzati viene fornita come parte del programma Welcome.

COME COMUNICARE CON IL COMPUTER

A questo punto è stato dedicato un po' di tempo semplicemente alla battitura da tastiera e all'osservazione dei risultati. In un primo momento sembra un fatto piuttosto ovvio che premendo, ad esempio, "A", il computer visualizzi una A sullo schermo. In effetti, occorre che una parte essenziale del computer, denominata **sistema operativo** (MOS), sia incessantemente attiva solo per produrre un risultato apparentemente semplice come quello dell'esempio precedente. In realtà, il MOS funziona per tutto il tempo che il computer rimane acceso. Anche quando il computer sembra essere inattivo, in "attesa" che si batta qualcosa da tastiera, il MOS opera per mantenere la visualizzazione sullo schermo e per eseguire altre funzioni vitali.

Il MOS ha anche il compito di richiamare gli altri sistemi forniti dal computer, ad esempio VIEW. Può essere attivo solo un sistema alla volta e, a meno che si disponga altrimenti, il MOS selezionerà automaticamente il sistema in linguaggio BASIC quando il computer è acceso: è questo il motivo per cui nella visualizzazione su schermo che compare a pagina 7 appare la parola BASIC. In seguito, tutto l'input da tastiera viene raccolto dal MOS e passato al sistema prescelto. Nella precedente sezione si sono impartite, probabilmente senza esserne consapevoli, istruzioni in BASIC. I messaggi visualizzati, come ad esempio "Mistake" o "Missing", sono stati prodotti dal sistema BASIC per indicare che non era in grado di decifrare la riga introdotta. Ovviamente, è stato il MOS ad eseguire l'ordine di disporre i caratteri sullo schermo.

Vi sono, ad ogni modo, circostanze che richiedono una comunicazione diretta con il MOS, indipendentemente dal sistema correntemente in uso. Questi comandi di sistema operativo hanno, come primo carattere, un asterisco (*) e questo simbolo viene usato per comunicare al MOS di occuparsi anche della parte rimanente della riga.

Per esempio, se si batte:

*HELP RETURN

il MOS darà in risposta una lista dei nomi e dei numeri di versione dei programmi contenuti nel computer.

*ROMS RETURN

farà in modo che il MOS elenchi i vari sistemi e linguaggi presenti nella ROM del computer. Il PC 128 S contiene il BASIC ed il software del sistema di archiviazione ADFS.

Provare ora a battere:

*BASIC RETURN

Il MOS riavvia il linguaggio BASIC, mentre sullo schermo viene visualizzata la parola BASIC.

Il comando *KEY fa in modo che il MOS associ una sequenza di caratteri ad un particolare tasto funzione. Per esempio, se si batte:

*KEY0funzione RETURN

ciascuna pressione successiva di f0 produrrà la scritta "funzione".
È così possibile, per esempio, abbreviare l'input della frase "i tasti funzione hanno una funzione" battendo:

i tasti f0 hanno una f0

In questo esempio abbastanza banale, la riga rimane incompleta (vale a dire, è possibile aggiungere ulteriori caratteri, cancellare caratteri ecc.) esattamente come se i caratteri venissero battuti uno alla volta dalla tastiera.

Altri comandi di sistema operativo possono essere usati per comunicare al MOS di modificare il suo funzionamento. Si ricorderà, ad esempio, che tenendo premuto un tasto della tastiera si introduce la funzione di *auto–ripetizione*, con la quale il carattere, dopo un breve lasso di tempo, viene ripetuto. Sia il breve lasso di tempo iniziale sia la velocità alla quale il carattere è ripetuto sono controllati dal MOS e se necessario possono essere cambiati. Riesaminare le normali impostazioni producendo una sequenza di caratteri con l'auto–ripetizione (e RETURN), quindi battere:

*FX12,1 RETURN

e ripetere la sequenza. Ora vedere che cosa succede se si cerca di produrre la stessa sequenza dopo aver battuto:

*FX11,0 RETURN

In altre parole, *FX12 consente di regolare la velocità alla quale i caratteri vengono ripetuti e *FX11 permette di regolare il lasso di tempo che intercorre prima dell'auto–ripetizione (*FX11,0 in realtà disattiva completamente la funzione di auto–ripetizione). E' possibile ripristinare le impostazioni iniziali della velocità e del lasso di tempo battendo:

*FX12,0 RETURN

Un riassunto di questi, e dei tantissimi altri effetti speciali è fornito nell'Appendice 5.

Infine il MOS "risponde" direttamente alle pressioni dei tasti controllo, come ad esempio CTRL + G e CTRL + L menzionati in precedenza. Questi due esempi implicano un effetto immediato, ma altri, come ad esempio CTRL + S (che può essere usato per modificare i colori dello schermo nei modi da 0 a 6 e da 128 a 134) richiedono ulteriori pressioni di tasto per avere effetto. Selezionare, per esempio, il modo 3 e premere:

CTRL + S seguito da 0 4 0 0 0

I cinque caratteri addizionali non compaiono sullo schermo ma il MOS li interpreta come una richiesta di cambiare in blu (4) il colore di sfondo (0). Analogamente:

CTRL + S seguito da 7 1 0 0 0

modifica in rosso (1) il colore del testo (7).

CTRL + T, o un successivo cambio di modo reimposta lo schermo nei suoi valori di default di testo bianco su sfondo nero.

USO DEI DISK DRIVE

Il PC 128 S è fornito di un disk drive situato nell'unità a disco del computer. In aggiunta, il PC 128 S può essere corredato opzionalmente da un secondo disk drive, collocabile anch'esso nell'unità a disco del PC 128 S. I disk drive del PC 128 S servono a leggere/scrivere dati su dischi magnetici da 3,5 pollici e sono quindi usati per archiviare dati e programmi per un uso successivo.
Il disco Welcome fornito con il computer contiene già tutta una serie di programmi, assai potenti ma allo stesso tempo semplicissimi da utilizzare, per il PC 128 S. Il capitolo successivo illustrerà come iniziare ad utilizzare questi programmi.

2. Utilizzo dei programmi Welcome ad icone

Il personal computer PC 128 S è fornito, sul disco Welcome, di un insieme di programmi ad icone. Questi programmi creano un ambiente "tavolo di lavoro" simulato che in pratica funziona come alcuni computer largamente utilizzati ma molto più costosi. La grande potenza ed utilità di questi programmi ad icone verrà presto constatata dall'utente che ne farà uso.

Caricamento dei programmi Welcome

Assicurarsi che il disco Welcome sia inserito nel drive 0, che nella configurazione base del PC 128 S è l'unico drive presente. Quindi tenere premuto SHIFT + BREAK, rilasciando per primo il tasto BREAK.

A questo punto la spia luminosa del drive si accenderà e dopo alcuni secondi il contenuto dello schermo scomparirà per visualizzare il menu principale Welcome, mostrato in seguito, composto da una riga di parole nella parte superiore e da un cursore a freccia.

Funzionamento del menu Welcome

La caratteristica principale di questo tipo di schermata, conosciuta come schermata ad icone, è che gli ordini impartiti al computer non vengono battuti da tastiera, ma trasmessi semplicemente indicando un oggetto presente sullo schermo. Gli "oggetti" in questione possono essere parole o piccole figure denominate *icone*.

Ovviamente, non è sufficiente limitarsi a toccare con le dita il simbolo appropriato. E' necessario, invece, utilizzare un puntatore per comunicare al computer che cosa si sta indicando esattamente. Esistono due modi per eseguire questa operazione:

– In primo luogo, si possono utilizzare i tasti cursore dislocati nella parte destra della tastiera principale. Se si tiene premuto uno di questi tasti, si vedrà il piccolo cursore a freccia spostarsi sullo schermo nella direzione voluta. Inoltre, se si continua a tenere premuto il tasto, il cursore si muoverà più velocemente, riducendo in tal modo il tempo impiegato per attraversare lo schermo.

– In secondo luogo, per spostare il cursore attraverso lo schermo si può utilizzare un joystick o un mouse con un connettore adeguato. Logicamente, spingendo il joystick (o facendo scorrere il mouse) verso destra il cursore sullo schermo si sposterà a destra, e così via. Per informazioni su come collegare questi dispositivi al computer, consultare l'Appendice 1.

Qualsiasi metodo si utilizzi, una volta posizionato il cursore nel punto desiderato occorre comunicare al computer che si desidera impartire un comando. Questa operazione è denominata *selezione* dell'oggetto e viene eseguita dalla tastiera per mezzo del tasto RETURN. Se si utilizza un joystick o un mouse, un pulsante particolare, generalmente il pulsante "fuoco", avrà lo stesso effetto.

Quasi tutte le funzioni di una schermata a icone possono essere controllate da questa azione combinata di "puntare e premere", che è comunemente denominata "*attivazione*". Provare a spostare il cursore verso la parte superiore dello schermo e posizionarlo sulla voce contrassegnata "Applicazioni". Si noterà che quando si sposta il cursore sul nome e si preme il pulsante o il tasto RETURN, sotto il cursore appare una breve lista di opzioni, denominata *menu* e la riga delle voci su cui si selezionano i menu viene denominata *Riga di menu*. Far scorrere il cursore lungo la Riga di menu e notare i vari menu che appaiono al di sotto di ciascuna voce. Se si sposta il cursore lungo uno qualsiasi dei menu, ogni voce indicata dal cursore verrà evidenziata.

Ritornare ora alla voce "Applicazioni" sulla Riga di menu, attivarla e spostarsi verso il basso alla voce "Scrivania". Se si seleziona quest'ultima premendo RETURN, lo schermo visualizza la riga di menu "Scrivania".

Spostare il cursore su "Schedario", attivarlo e spostarsi sul primo elemento di menu. Selezionando quest'ultimo, apparirà un riquadro contrassegnato "Schedario" contenente un certo numero di parole e di simboli. Si tratta della *finestra* di "Schedario" e contiene icone per tutte le funzioni associate al programma "Schedario". E' possibile spostare la finestra sullo schermo posizionando il cursore sul riquadro del titolo "Schedario" della finestra, attivandolo e quindi spostando il cursore nella posizione desiderata e ripremendo il pulsante. E' anche possibile eliminare completamente la finestra dallo schermo, selezionandola e poi premendo il tasto DELETE.

SCHEDARIO

Schedario è un utile programma di data base, che memorizza le informazioni sotto forma di righe, chiamate *campi*, che sono inizialmente etichettate con Nome, Indirizzo, Tel. e Altro e che vengono visualizzate sullo schermo in modo simile a un vero schedario. Questa funzione può essere utile se si intende conservare un elenco nominativo di persone, ad esempio amici o colleghi di lavoro.

La prima voce del menu, "On/Off", controlla se la finestra contenente i record (schede) del file è visibile o meno. Un file è un blocco di informazioni, tipo archivio, che viene memorizzato su disco. Per prima cosa però, occorre comunicare a Schedario il nome del file nel quale dovranno essere memorizzati i record. Questa operazione è eseguita visualizzando nuovamente il menu Schedario e selezionando "Caricamento". A questo punto sullo schermo apparirà una finestra che propone un nome per il file. Il computer attenderà la pressione di RETURN. Se si desidera utilizzare un altro file, con un nome diverso, è necessario selezionarlo dal programma "Catalogo" (vedere più avanti per dettagli). Una volta scelto il file, la finestra scomparirà e dopo una breve pausa i campi della finestra si riempiranno per mostrare il primo record del file.

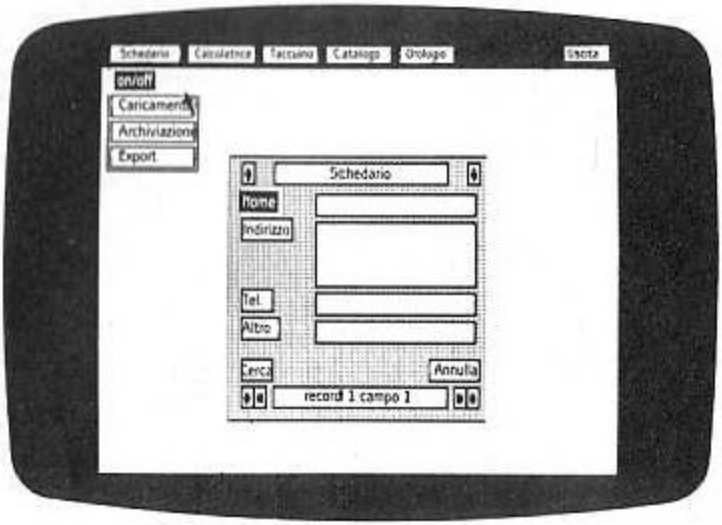
La prima volta che si seleziona la finestra Schedario quest'ultima sarà vuota, consentendo così all'utente di creare un proprio schedario semplicemente riempiendo le righe vuote di ciascun record. Per fare ciò è necessario portare il cursore a freccia sulle righe vuote (campi) e digitare sulla tastiera del computer. Quando sono state introdotte le informazioni, o dopo aver apportato delle modifiche su uno schedario preesistente, visualizzare il menu Schedario e selezionare "Archiviazione". Questa operazione salverà lo schedario sul disco, in modo che possa essere utilizzato in futuro. Il computer riproporrà un nome per il file che potrà essere accettato premendo RETURN o modificato utilizzando DELETE per cancellare il nome proposto e introducendo quindi il nome voluto. Il disco Welcome non deve essere utilizzato per questo scopo ed è consigliabile, prima di eseguire questa operazione, consultare la sezione "Sistemi di archiviazione".

Per spostarsi ad altre schede si possono attivare le icone, riportate sul fondo dello schedario, delle frecce verso sinistra e verso destra o doppia freccia: le prime consentono di muoversi attraverso lo schedario di una scheda alla volta, le seconde di dieci schede alla volta.

Le frecce riportate a fianco del titolo "Schedario" permettono di cambiare campo.

La modifica delle informazioni sulle schede è molto semplice. Puntare la riga da modificare e battere il nuovo dato, usando DELETE se si commettono degli errori.

Per ricercare una scheda particolare, occorre per prima cosa spostare il cursore sulla riga appropriata (Nome, Indirizzo, ecc.) e quindi attivare l'icona 'Cerca'. A questo punto apparirà una finestra, che richiede il testo da ricercare. Dopo avere introdotto la risposta e avere premuto RETURN il programma controllerà ciascuna scheda, confrontandola con il testo introdotto. La ricerca è condotta in modo abbastanza versatile; è possibile battere semplicemente una singola lettera o diverse parole e se viene incontrata una scheda contenente il testo nel campo selezionato, questa verrà visualizzata. Ulteriori schede interessate possono essere vi-

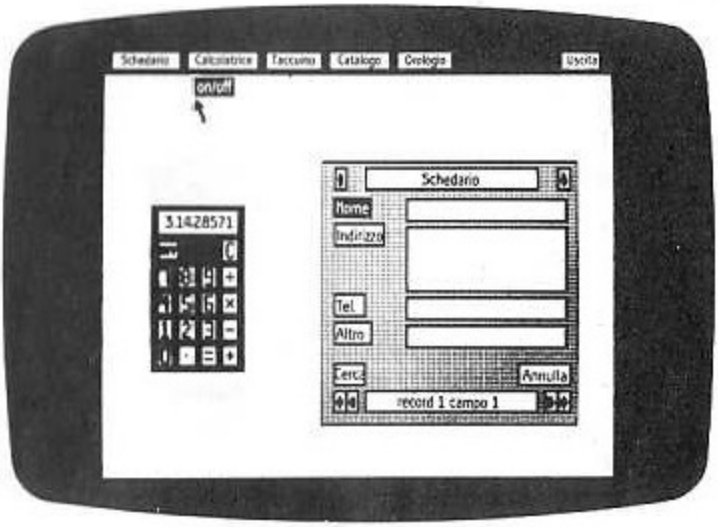


sualizzate attivando le icone delle frecce sinistra e destra, mentre 'Annulla' consentirà di visualizzare di nuovo l'intero file (e non semplicemente le schede contenenti il testo ricercato).

Schedario ha una caratteristica particolare, che consente di incorporare informazioni in esso contenute all'interno di documenti preparati con il sistema di word processing VIEW. L'ultima voce nel menu Schedario, 'Export', permette il trasferimento dei nomi e degli indirizzi dello schedario in VIEW, dove possono essere utilizzati per la preparazione di lettere o per altri scopi. Questa funzione è descritta in dettaglio nella sezione riguardante VIEW, più avanti in questo manuale.

CALCOLATRICE

La seconda voce sulla riga di menu è "Calcolatrice", e cioè una calcolatrice a pulsanti che possono essere attivati dopo averli puntati col cursore a freccia. Il menu dispone di una sola voce "On/Off", che controlla se la calcolatrice è visibile o meno. Se si seleziona questa voce, si vedrà che dispone di tutti i tasti di una normale calcolatrice a quattro funzioni. Quando il puntatore si trova sulla calcolatrice, si possono anche utilizzare i tasti equivalenti sulla tastiera, incluso 'C' per cancellare il display e 'O' per disattivarlo (ossia cancellarlo dallo schermo). La calcolatrice può essere spostata sullo schermo selezionando l'area grigia sotto la finestra del display.



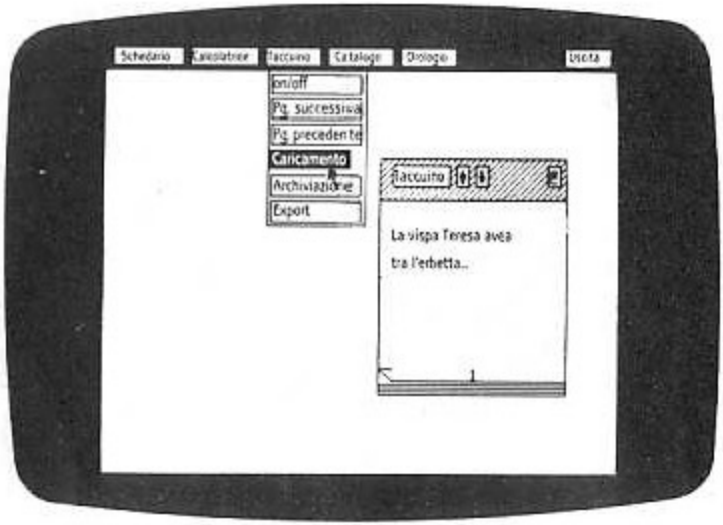
TACCUINO

Il Taccuino consente di eseguire elettronicamente brevi annotazioni; memorizza ciò che viene battuto in una propria area di memoria, in tal modo può essere utile come agile "memorandum". Spostando il cursore sulla voce Taccuino sulla riga di menu si produce una breve lista di voci, la prima delle quali "On/Off", consente di controllare se il Taccuino è visibile o meno. Se si seleziona questa voce, il Taccuino viene visualizzato sullo schermo mostrando la prima pagina di testo (che inizialmente è vuota). Le icone delle frecce verso l'alto e verso il basso, e le voci equivalenti del menu, consentono di girare una a una le pagine del blocco. Ciascuna pagina può memorizzare fino a 20 righe di 31 caratteri, e il numero massimo di pagine è 16.

Il Taccuino dispone di una icona che non è stata incontrata in precedenza, una sorta di 'L' invertita chiamata *riquadro di dimensionamento* che permette di espandere e ridurre le dimensioni della finestra in un modo simile allo spostamento della stessa. Provare a verificarne il funzionamento. Il Taccuino viene spostato operando una selezione sul titolo nella parte superiore della finestra.

E' possibile memorizzare il Taccuino su disco o recuperare il Taccuino già memorizzati per mezzo delle voci 'Caricamento' e 'Archiviazione' del menu, che richiederanno l'introduzione di un nome di file, come sempre avviene in questi casi.

L'ultima voce del menu del Taccuino, 'Export', permette il trasferimento del contenuto del Taccuino nel word processor VIEW, in modo che possa essere stampato. Questa funzione è descritta in modo più dettagliato nella sezione riguardante VIEW, più avanti in questo manuale.



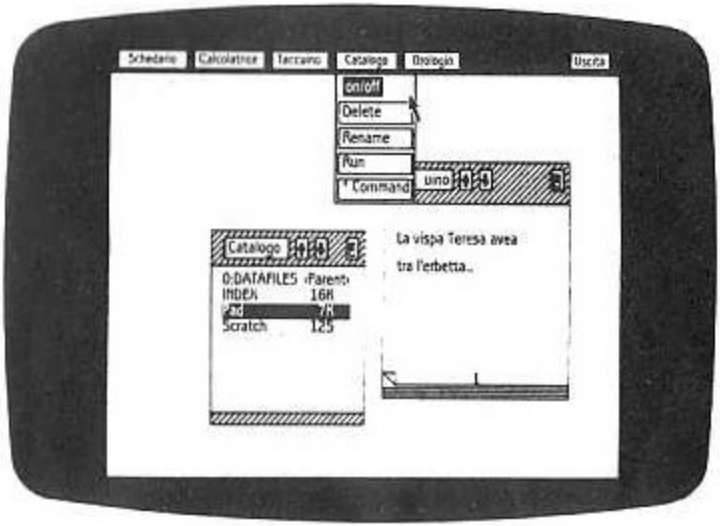
CATALOGO

La funzione 'Catalogo' consente di esaminare l'indice del disco che si trova nel drive. I programmi e le informazioni vengono memorizzati sul disco in blocchi, denominati **file**, e questi file sono organizzati in gruppi, denominati **directory.** Fare il "Catalogo" di una directory significa visualizzare quali file contenga. Per meglio comprendere ciò che segue, consultare il capitolo sui sistemi di archiviazione.

Non appena si punta la voce 'Catalogo' sulla riga di menu, appare un menu contenente una voce che permette di attivare e di disattivare la finestra Catalogo, e tre altre voci. Se si seleziona la prima voce apparirà la finestra Catalogo contenente una lista dei file e delle sub–directory della directory correntemente selezionata (CSD). Nella parte superiore della lista si trova la voce '<Parent>', che indica l'indice al di sopra della CSD nella struttura del disco. Al di sotto di questo vi sono voci ordinate alfabeticamente che indicano sub–directory (contrassegnate '<Dir>') e file, seguiti dalle loro dimensioni espresse in byte o Kilobyte (indicati da una 'K'). Si può espandere la finestra con il riquadro di dimensionamento per prendere visione di altre informazioni riguardanti i file.

La selezione di una voce della lista produce un effetto diverso a seconda del tipo di voce: la selezione di una directory consentirà di visualizzare il suo contenuto, mentre la selezione di un file ne provocherà l'evidenziazione, in attesa di un ulteriore comando. Questi comandi possono essere costituiti da una qualsiasi delle tre voci rimanenti del menu, che sono funzioni standard ADFS i cui effetti verranno trattati in dettaglio nel capitolo sui sistemi di archiviazione.

Un'ulteriore voce del menu è "*COMMAND." che serve per eseguire direttamente qualsiasi comando del sistema operativo o del sistema di archiviazione (vedasi i capitoli successivi sugli argomenti suindicati).



OROLOGIO

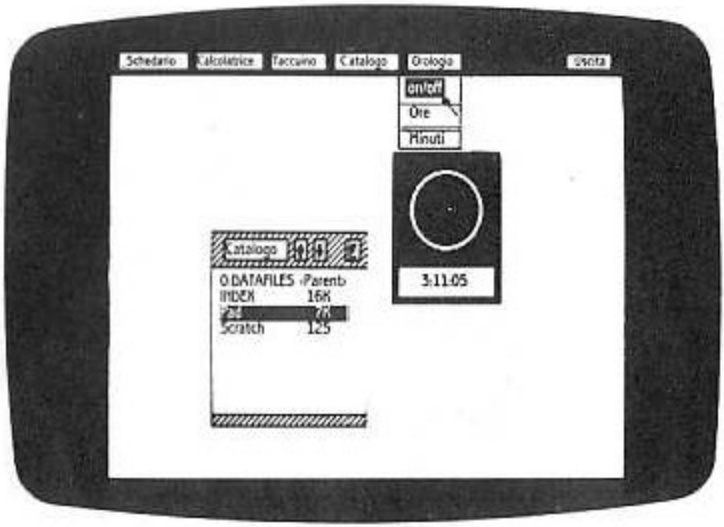
Attivando la prima voce del menu Orologio viene visualizzato un orologio analogico digitale, perfettamente funzionante. Se lo si desidera, l'orologio può essere spostato sullo schermo puntando il cursore in qualsiasi parte dell'orologio stesso o impostato usando le voci 'Ore' e 'Minuti' nel menu. Per fare ciò premere RETURN con il cursore a freccia puntato sulle voci "Ore" e "Minuti" rispettivamente. Nonostante il suo ottimo funzionamento, l'orologio in questione potrebbe fermarsi se si eseguono altre operazioni in finestre differenti. Ad ogni modo, se si sposta di nuovo il cursore nella finestra 'Orologio', l'ora indicata si imposterà automaticamente. Potete accelerare la selezione di ore/minuti premendo il tasto SHIFT + RETURN. Premendo – o DELETE si decrementa l'ora.

USCITA

Il menu 'Uscita' fornisce un modo rapido per ritornare al menu iniziale incontrato all'inizio di questa sezione. Non contiene altre voci. Quando selezionato vi verrà richiesta conferma. Premendo RETURN ritornerete al menu principale.

Altri programmi Welcome

Torniamo ora al menu principale Welcome per esaminare altri programmi dimostrativi. Spostare il puntatore sul menu 'Uscita' e attivarlo. Dopo alcuni istanti sullo schermo appare la pagina del menu principale Welcome contenente una fila di menu visualizzata nella parte superiore dello schermo. I menu raggruppano i programmi a seconda della loro funzione. Esaminiamo ora i programmi grafici, raggruppati sotto il titolo "Grafica".



Grafica

CASTELLO illustra la capacità del computer di produrre una grafica multicolore ad alta velocità. Utilizza molteplici figure geometriche (quadrati, rettangoli, cerchi e triangoli) a tinta unita o con motivi fantasia.

CLOWN è simile a CASTELLO, ma utilizza altre forme.

NUVOLA è una semplice sequenza animata nella quale varie parti di un'immagine grafica vengono spostate sullo schermo. La scorrevolezza del movimento si ottiene passando dagli schermi normali agli schermi ombra (*shadow*). Per produrre un particolare effetto, si possono utilizzare cambiamenti di colore.

DISEGNI produce una sequenza di figure complesse, dalla piacevole resa estetica, che sono destinate ad illustrare la velocità con cui il computer può riempire un'area con un colore uniforme o con un modello più complesso.

FIGURE è una sequenza di esempi che mostra le figure basilari che possono essere prodotte direttamente usando comandi grafici incorporati. A scopo dimostrativo, ogni figura è disegnata come figura piena, ed è sovrapposta alla precedente, ma è anche possibile produrre solo il contorno delle figure servendosi di linee continue o tratteggiate.

Tutorials

Il programma esplicativo su ADFS Advanced Disk Filing System (sistema avanzato di archiviazione su disco) rappresenta un'introduzione al modo in cui il computer memorizza i dati su disco. Il capitolo 6 di questo manuale è interamente dedicato allo ADFS.

TASTIERA è un programma progettato per facilitare l'apprendimento delle operazioni effettuabili da tastiera. TASTIERA visualizza un carattere. Il tasto che rappresenta questo carattère deve essere trovato sulla tastiera e premuto. Il computer calcola il tempo impiegato e visualizza il punteggio ottenuto (max 10), in ognuno dei cinque test diversi e il tempo medio impiegato per trovare ciascun tasto.

MODES passando attraverso gli otto modi di visualizzazione di base visualizza esempi di testo, i colori disponibili e le funzioni grafiche di base, se ammesse.

TESTO visualizza il testo in diversi modi e stili.

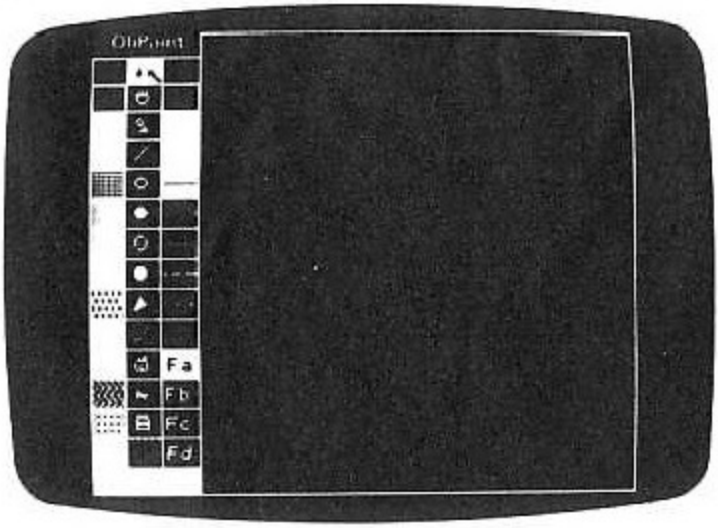
Applicazioni

OLIPAINT

Questo programma consente di creare e di salvare le figure create utilizzando molte delle caratteristiche grafiche avanzate fornite dal computer. Inoltre consente di cambiare i colori di schermo del PC 128 S.

Una volta caricato, OLIPAINT visualizza lo schermo riportato nella prossima figura.

L'area sulla sinistra è il menu, dal quale si selezionano le varie funzioni, colori e caratteri di stampa (tipo di caratteri) che si desidera utilizzare. L'area più ampia, vuota, sulla destra è la "tela" su cui vengono create le immagini.



Il menu è diviso in 3 colonne:

- La colonna posta all'estrema sinistra determina quale colore verrà utilizzato per tutte le operazioni successive; una volta selezionato, questo colore apparirà nel rettangolo più grande posto nella parte inferiore del menu;

- La colonna di mezzo contiene tutte le funzioni disponibili, ognuna delle quali è identificata da un simbolo speciale, come una macchina fotografica, una macchina per scrivere, un paio di forbici ecc; ognuno di questi simboli verrà descritto in seguito;

- La colonna posta all'estrema destra è ulteriormente suddivisa:
  - le quattro caselle in alto contengono la tavolozza dei colori;
  - le caselle immediatamente al di sotto della tavolozza contengono i sei tipi di riga che possono essere utilizzati;
  - le quattro caselle in basso mostrano che tipo di caratteri di stampa verrà utilizzato quando il testo viene disposto sullo schermo;
  - l'ultima casella in basso rappresenta l'opzione "Griglia".

Le selezioni dal menu vengono eseguite spostando il puntatore a freccia nella casella richiesta e premendo CTRL. I quattro tasti di controllo cursore forniscono un movimento lento, mentre se premuti simultaneamente a SHIFT forniscono un movimento rapido (otto volte il normale) (è inoltre possibile azionare OLIPAINT utilizzando un joystick. In questo caso il pulsante "fuoco" sostituisce la funzione di CTRL. Vedere l'Appendice 1 per istruzioni su come collegare un joystick al computer).

Le selezioni del menu vengono evidenziate e, quando OLIPAINT viene caricato inizialmente, le selezioni sono:

- sfondo nero;
- primo piano bianco;
- opzione "spray gun (vedere più avanti);
- joystick disattivati;
- griglia disattivata;
- linee continue;
- tipo di carattere normale (Fa).

Per selezionare uno sfondo di colore diverso dal nero, spostare il puntatore sulla casella contenente il colore o il modello prescelto e premere CTRL; nella casella posta nella parte inferiore del menu apparirà la scelta effettuata.

E' possibile modificare la tavolozza (vale a dire la gamma delle combinazioni di colore disponibili) utilizzando le quattro caselle poste nella parte superiore della terza colonna del menu. Per esempio, per modificare il rosso in verde spostare il puntatore nella casella rossa e tenere premuto CTRL finchè nella casella non appare il colore verde.

Analogamente, la funzione di tracciamento e i vari stili di carattere e di tipi di linea vengono selezionati spostando il puntatore nella casella appropriata e premendo CTRL.

Il caricamento e la memorizzazione vengono eseguiti premendo L o S. Questa operazione cancella il contenuto dell'area di menu e consente di battere il nome che si desidera utilizzare.

Per cancellare il contenuto dello schermo e cominciare di nuovo, premere il tasto "R". Ciò reimposterà tutte le opzioni ai loro valori iniziali.

Ogni qualvolta si sposta il puntatore al di fuori dell'area di menu la freccia viene sostituita con il simbolo che indica la funzione selezionata. Le varie procedure vengono descritte in seguito, partendo dal presupposto che sia stata già selezionata dal menu la funzione corrispondente.

Lo "spray gun" (spruzzatore) consente di tracciare una o più linee per volta; ogni punto dello "spruzzo" lascia una linea dietro di sè quando si sposta. Premere COPY per aumentare il numero di punti e DELETE per ridurlo. Per usare lo "spray gun", spostarsi nella posizione dove si vuole iniziare a tracciare la linea e premere CRTL quindi, tenendo premuto CTRL, spostarsi attraverso lo schermo e rilasciare CTRL.

Il simbolo della mano consente di spostare l'intero schermo in qualsiasi direzione. Per cominciare, spostare la mano di un punto dello schermo facilmente identificabile, quindi premere CTRL. Tenendo premuto CTRL, spostare la mano nella posizione nella quale si desidera spostare il punto originale e rilasciare CTRL. L'intero schermo si sposterà. Notare che qualsiasi parte dell'immagine scompaia dallo schermo non potrà essere in seguito recuperata.

L'opzione "flood fill" (riempimento) può essere usata per riempire con il colore corrente qualsiasi area chiusa dello schermo. Spostarsi semplicemente in qualsiasi punto all'interno dell'area che si desidera riempire e premere CTRL. Notare che se si cerca di riempire un'area il cui contorno presenta delle interruzioni, il colore fuoriuscirà dall'apertura e non si arresterà finchè non incontrerà un confine chiuso o i bordi dello schermo. Le aree possono essere svuotate del colore premendo TAB. Talvolta, però, questa operazione non solo inverte l'effetto della funzione di riempimento, ma interessa anche le altre figure della "tela", a seconda dei colori utilizzati.

La funzione "linea" consente di tracciare linee singole in qualsiasi punto della tela. Premere CTRL per iniziare a tracciare una linea e rilasciarlo. Disegnare ciò che si vuole coi cursori. Premere e rilasciare CTRL per fissare l'immagine.

Il contorno di un'ellisse o un'ellisse piena vengono prodotti spostandosi al punto destinato ad essere il centro dell'ellisse e premendo CTRL. Quindi, tenendo premuto CTRL, può essere modificata l'ampiezza dell'ellisse. Rilasciare CTRL e servendosi dei tasti freccia si predispone l'altezza. Premere CTRL per ottenere l'ellisse.

Circonferenze e cerchi pieni vengono disegnati in modo simile. Premere CTRL per indicare il centro e modificare la posizione del simbolo circolare fino a produrre le dimensioni desiderate, quindi rilasciare CTRL.

L'opzione "macchina fotografica" consente di realizzare copie di qualsiasi area rettangolare della tela. Spostare il simbolo in uno degli angoli dell'area, purché esterno o coincidente con essa, mai interno, e premere CTRL. Quindi tenere premuto CTRL, spostando contemporaneamente i tasti cursore per riquadrare la profondità e la larghezza del rettangolo. Rilasciare CTRL. La pressione dei tasti cursore consentirà ora lo spostamento di una seconda casella, che dovrebbe essere posizionata nel punto in cui si desidera collocare la copia. La copia viene eseguita premendo CTRL.

L'opzione "forbici" consentirà lo spostamento di un'area rettangolare dello schermo, sostituendola con un blocco dello stesso colore dello sfondo. Questa operazione viene eseguita nello stesso modo della routine di copiatura descritta in precedenza.

L'opzione "triangolo" consente di tracciare dei triangoli, ognuno dei quali ha inizio in un punto identificato dalla pressione di CTRL. Tenendo premuto CTRL ci si muove al secondo angolo e si rilascia il tasto. Infine ci si muove al terzo angolo coi cursori a freccia e lo si fissa premendo e rilasciando CTRL.

Per disegnare "archi", "segmenti" e "settori" ci si muove nel punto centrale del cerchio cui si vuole che appartengano e si preme CTRL. Tenendo premuto CTRL si traccia il raggio che interessa. Si rilascia CTRL e ci si muove fino ad ottenere l'angolo desiderato. Premendo e rilasciando CTRL si fissa l'immagine sullo schermo.

L'opzione "macchina per scrivere" può essere usata per stampare un testo sullo schermo. Sulla tela, si identifica il punto iniziale del testo. Premendo CTRL + R, il simbolo della macchina per scrivere viene sostituito da una matita.
Qualsiasi successiva pressione di tasto produce sullo schermo caratteri del tipo corrente e la fine del testo è contrassegnata premendo RETURN. Notare che il testo può essere collocato tra il punto di inizio e la parte destra della tela. Non è possibile usufruire del ritorno a capo automatico.

L'opzione "griglia" riduce il movimento dei simboli alle posizioni di una immaginaria "griglia" posta sullo schermo, rendendo così più semplici operazioni come tracciare circonferenze di raggio diverso con lo stesso centro ecc.

Quando viene selezionata, l'opzione "joystick" consente di controllare il movimento dei simboli tramite un joystick invece che premendo i tasti freccia. Inoltre il pulsante "fuoco" sostituisce il tasto CTRL.

ViewSheet e VIEW, che sono programmi applicativi, vengono trattati individualmente nei capitoli successivi di questo manuale o nei manuali specializzati interamente dedicati a questi programmi (es. manuali VIEW e ViewSheet).

Giochi

COBRA: nutrite il cobra senza mangiarvi la coda...

U-BOOT 87 è un gioco sofisticato nel quale si totalizzano punti distruggendo vari bersagli in uno scenario sottomarino. Usando la tastiera oppure un joystick si controlla il movimento del sommergibile nero (vedere l'Appendice 1 per istruzioni sul collegamento dei joystick).
I punti vengono calcolati in base ad ogni colpo andato a segno ai danni di un bersaglio, ma alcuni di questi ultimi, per essere distrutti, devono essere colpiti più di una volta. Se si colpisce il serpente marino, questo si divide in due segmenti. Se si entra in collisione con un bersaglio, con la piovra o con una mina vagante, si perde una delle tre "vite" di cui si dispone.

Programmi di Utilità

Il programma DEF. CARATTERI permette di ridefinire i caratteri visualizzati sullo schermo dal computer in modo da produrre effetti simili a quelli prodotti dal programma TESTO visto in precedenza.

GEN. SUONO è un programma che dimostra la versatilità del circuito generatore di suono. Con questo programma si può utilizzare la tastiera del computer come se fosse quella di un pianoforte. Fare riferimento alla figura riportata più avanti.

PAN. CONTROLLO è un programma che visualizza le impostazioni correnti delle opzioni di configurazione contenute nella EEPROM e che permette di modificarle. Gli effetti prodotti dal programma sono esattamente uguali a quelli prodotti dal comando MOS *CONFIGURE. Gli utenti esperti potranno fare riferimento all'appendice relativa ai comandi MOS alla fine di questo manuale. Si noterà che alcune delle informazioni fornite in questa sezione presuppongono la conoscenza di caratteristiche del computer che non sono state ancora esaminate. Tutto diventerà più chiaro proseguendo nella lettura di questo manuale.
Quando si seleziona il pannello di controllo (Control Panel) dal menu Utilità, il programma visualizza una serie di figure sullo schermo e diverse informazioni in punti diversi dello stesso. Spostare il puntatore sullo schermo e modificare le opzioni di configurazione selezionando all'interno dei riquadri. Notare però che queste modifiche non saranno effettive finchè non verranno memorizzate, come descritto di seguito.

Il personal computer PC 128 S contiene una memoria chiamata EEPROM o Status RAM, che mantiene in memoria i dati anche dopo lo spegnimento del computer. Questa memoria viene utilizzata per memorizzare informazioni sulla configurazione di varie opzioni,da qui il nome di Status RAM, relative al funzionamento del computer.

PATTERN EDITOR permette di creare propri modelli da utilizzare con il comando PLOT che verrà illustrato più avanti in questo manuale, nella sezione relativa al BASIC.

ADFS mette a disposizione diversi programmi per la gestione dei dischi, incluso il programma "S.D.BACKUP" per fare copia dei dischi. (Vedi Capitolo 6).

DEFINIZIONE DI CARATTERI

Questo programma permette di modificare la forma delle lettere e dei numeri visualizzati sullo schermo. In altre parole permette di creare propri tipi di carattere (font).

Lo schermo è suddiviso in tre aree. L'area superiore contiene le forme di carattere correnti. L'area centrale visualizza una versione ingrandita del carattere selezionato correntemente (con accanto la versione in dimensioni reali). L'area inferiore visualizza un riassunto dei tasti con funzioni speciali.

I caratteri da ridefinire possono essere selezionati in uno dei due modi seguenti:

- premendo il tasto appropriato da tastiera (per i caratteri normali);
- premendo DELETE e utilizzando i tasti di movimento cursore per posizionare il cursore al di sotto di uno dei caratteri visualizzati nella parte superiore dello schermo e premendo nuovamente DELETE per riposizionare il cursore all'interno dell'area centrale. Questo metodo può essere utilizzato per selezionare i caratteri standard ed i caratteri che non possono essere ottenuti direttamente da tastiera.

Dopo aver selezionato un carattere, nel riquadro centrale appare una versione ingrandita dello stesso. A questo punto possono essere utilizzati i tasti cursore per selezionare un particolare elemento del riquadro centrale. La pressione del tasto RETURN provoca il cambiamento di stato dell'elemento (cioè se l'elemento è bianco, RETURN lo trasforma in nero e viceversa). L'effetto delle modifiche si riflette immediatamente sul carattere a destra della griglia.

ESCAPE viene utilizzato per terminare l'esecuzione del programma.

COPY viene utilizzato per salvare sul disco il tipo di carattere corrente. In questo modo possono essere creati più tipi di carattere, tutti memorizzabili e quindi ricaricabili dal disco quando richiesto.

TAB viene utilizzato per ripristinare il tipo di carattere normale. Notare che a meno che TAB non venga usato prima del termine del programma, gli effetti delle modifiche rimarranno attivi fino allo spegnimento del computer o ad un'interruzione hard break (CTRL + BREAK).

GENERATORE DI SUONO

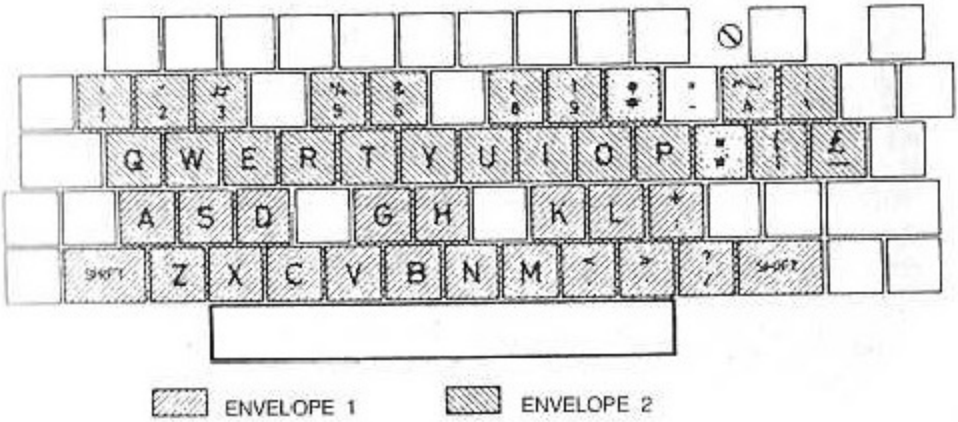
Il comando BASIC ENVELOPE verrà menzionato più avanti in questo manuale, ma il solo fatto che richieda un minimo di 14 parametri, lo rende impossibile da trattare in un manuale come questo. Ad ogni modo, questo è un programma di utilità che permette di utilizzare il comando di inviluppo. Esso permette di modificare alcuni o tutti i diversi parametri e di ascoltare l'effetto delle modifiche apportate ai suoni generati dal computer; quindi a valori diversi dei parametri corrisponderanno suoni diversi. Questo comando può inoltre essere utilizzato per determinare i parametri necessari per la creazione di un suono particolare da usare ad esempio come colonna sonora di un gioco.

Dopo il caricamento, il programma visualizza diversi riquadri. Due inviluppi diversi possono essere definiti modificando il contenuto del riquadro contrassegnato Number (ENVELOPE 1 - ENVELOPE 2). I riquadri rimanenti rappresentano le impostazioni dei 14 parametri associati con l'inviluppo selezionato correntemente:

Length	– Lunghezza di ogni passo: 1/100 sec.
Pstep1	– Cambi di tono di 1 in 1
Pstep2	– Cambi di tono di 2 in 2
Pstep3	– Cambi di tono di 3 in 3
Steps1	– Numero di passi nella sezione 1
Steps2	– Numero di passi nella sezione 2
Steps3	– Numero di passi nella sezione 3
AstepA	– Modifica dell'ampiezza nell'attack
AstepD	– Modifica dell'ampiezza nel decay
AstepS	– Modifica dell'ampiezza nel sustain
AstepR	– Modifica dell'ampiezza nel release
Peak	– Livello da raggiungere alla fine di attack
Level	– Livello da raggiungere alla fine di decay

Per spostarsi da un riquadro all'altro utilizzare ← e →. Il riquadro corrente viene evidenziato in nero. Per aumentare o diminuire il valore contenuto nel riquadro corrente premere rispettivamente ↑ oppure ↓. In alternativa, è possibile caricare diversi inviluppi preimpostati premendo un tasto funzione (f0 ... f9).

L'effetto dell'impostazione corrente dei parametri di ENVELOPE può essere ascoltato utilizzando la tastiera, che viene suddivisa in due tastiere simili alla tastiera del pianoforte:



Da uno o entrambi gli envelope si possono premere uno o più tasti.

E' possibile sostituire uno degli envelope preimpostati con uno proprio premendo COPY seguito dal numero di envelope da ridefinire.

PANNELLO DI CONTROLLO

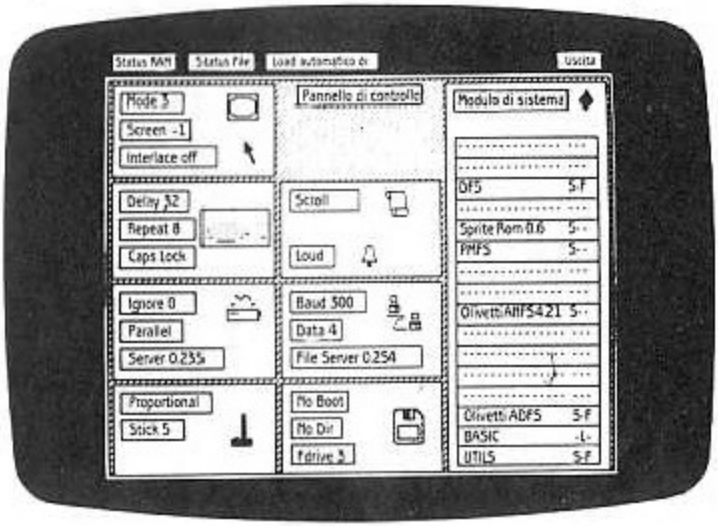
Osservando la visualizzazione prodotta dal Pannello di Controllo, si può notare, nell'angolo superiore sinistro, un riquadro contenente la parola 'Mode' (Modo) seguita da un numero, che serve ad impostare il modo di schermo che sarà attivo all'accensione del computer. Il modo può essere modificato attivando, cioè puntando col cursore a freccia, un punto all'interno del riquadro "MODE" oppure utilizzando i tasti '+' e '–' sul tastierno numerico, rispettivamente per aumentare o diminuire il numero. Anche tutte le opzioni descritte di seguito possono essere modificate utilizzando questi due tasti per aumentarne o diminuirne i valori. Premendo + e – contemporaneamente al tasto SHIFT si aumentano o diminuiscono i valori in passi di dieci.

Al di sotto di 'Mode' è presente l'opzione 'SCREEN'. Questa opzione permette di spostare verso l'alto o verso il basso la posizione dell'immagine generata dal computer. SCREEN è equivalente al primo parametro del comando *TV. L'opzione 'Interlace' controlla se è stata attivata o meno la visualizzazione di un numero doppio di righe sullo schermo. Per modificare la visualizzazione è sufficiente attivarla o disattivarla.

Nel riquadro sottostante sono presenti tre opzioni di tastiera: 'Delay' (intervallo), 'Repeat' (ripetizione) e 'Caps Lock'. La prima opzione imposta il tempo, trascorso il quale, viene attivata la funzione di ripetizione automatica del tasto premuto (espresso in centesimi di secondo), la seconda imposta la velocità di ripetizione dei tasti e l'ultima controlla se la spia di Caps Lock è impostata su CAPS LOCK (caratteri alfabetici maiuscoli), SHIFT CAPS (scrittura dei caratteri rappresentati sulla parte superiore dei tasti) o NO CAPS (scrittura dei caratteri rappresentati nella parte inferiore dei tasti).

Il terzo riquadro contiene le opzioni di controllo stampante. Il carattere di riconoscimento stampante (modificato anche con *FX6) può essere impostato su IGNORE, NO IGNORE o IGNORE seguito da un numero (che è il numero del carattere da ignorare). Al di sotto di questo è presente il tipo di stampante selezionato che può essere SINK, PARALLEL, SERIAL (cioè con interfaccia RS232 opzionale), USER oppure ECONET. Infine l'ultima opzione controlla il numero di dispositivo di una stampante Econet. I tasti '+' e '–' influenzeranno solamente il numero dopo il punto. Per modificare il numero prima del punto premere CTRL contemporaneamente a '+' o '–'. Queste informazioni servono soltanto agli utenti di rete Econet.

Nella parte inferiore della fila sinistra sono presenti le opzioni Mouse/Joystick. La prima controlla se il MOS simula un joystick SWITCHED (attivato) o PROPORTIONAL (funzione utile quando si utilizzano giochi). La seconda, STICK seguita da un numero, imposta l'accelerazione utilizzata dal MOS quando viene selezionato un joystick proporzionale.



Nella parte superiore della colonna centrale vi sono due opzioni. La prima, SCROLL, oppure NO SCROLL, controlla l'azione del software a finestre MOS e non dovrà essere presa in considerazione. La seconda, LOUD o QUIET imposta il volume totale dei suoni prodotti dall'altoparlante interno del computer.

Il secondo riquadro contiene le opzioni che controllano l'interfaccia seriale opzionale RS232. BAUD seguito da un numero imposta la velocità dell'interfaccia, mentre DATA seguito da un numero imposta il formato dei dati. L'opzione File Server imposta il numero di dispositivo di un File Server Econet. I tasti '+' e '–' influenzeranno solo il numero dopo il punto. Per modificare il numero prima del punto, premere CTRL contemporaneamente a questi tasti. Questa funzione è applicabile solo utilizzando la rete Econet.

Il riquadro inferiore contiene le opzioni che controllano l'ADFS ed i disk drive. BOOT o NO BOOT controllano se si è verificato un tentativo da parte di ADFS di caricare automaticamente un programma da disco quando la macchina è stata accesa o reinizializzata con il tasto BREAK. La seconda opzione, DIR o NO DIR, ha effetto se la directory viene reimpostata automaticamente quando il disco viene installato. Per ulteriori informazioni consultare la sezione riguardante i Sistemi di Archiviazione. L'ultima opzione, FDRIVE, seguita da un numero, imposta i parametri che controllano il funzionamento dei disk drive.

Questi parametri dovrebbero essere modificati soltanto da utenti esperti. La variazione di ciascuno di questi parametri altera lo stato del sistema in modo permanente. Pertanto si consiglia agli utenti non ancora esperti di non cambiare alcun parametro.

L'intera colonna destra del Pannello di Controllo rappresenta una tabella del software contenuto al momento nel computer. La tabella è una lista con 16 spazi, ciascuno dei quali rappresenta il contenuto di un chip ROM o di un'area RAM. Il nome di ogni parte di programma compare nella tabella ed è seguito da una combinazione di tre lettere o trattini. La lettera 'S' indica che il programma specificato è una ROM di Servizio, la lettera 'L' che si tratta di un programma di Linguaggio e la lettera 'F' che si tratta di un sistema di archiviazione. In alternativa, l'ultimo spazio può essere occupato dalla lettera 'U' che indica che la ROM non è collegata.

Sulla riga di menu nella parte superiore dello schermo sono presenti tre elementi di menu. Il primo 'Status RAM' contiene quattro opzioni. 'Originale' carica le impostazioni di opzione di default dal disco allo schermo, permettendo di riportare il computer allo stato iniziale. 'Aggiorna' permette di salvare nella EEPROM le proprie configurazioni di opzioni. Selezionando 'Aggiorna', sullo schermo appare un riquadro che richiede di confermare la scelta effettuata prima del salvataggio delle opzioni. 'Esci' permette di uscire dal programma Pannello di Controllo e di tornare al menu Icone. 'Ricomincia' legge le impostazioni di opzione correnti dalla EEPROM e le invia allo schermo in modo da annullare le modifiche apportate. Naturalmente, se si fosse già utilizzato 'Aggiorna' per salvare le impostazioni, questa opzione non avrà effetto.

Il secondo elemento di menu, 'Status File', permette di caricare ('CARICA') e salvare ('ARCHIVIA') le opzioni di configurazione sul disco (invece che nella EEPROM). E' possibile conservare su disco una serie di opzioni utilizzate più raramente.

L'ultimo elemento, 'Load automatico' controlla quale linguaggio ROM viene utilizzato e quale ROM di sistema di archiviazione viene eseguita dal computer all'accensione. Il menu comprende due elementi: 'Linguaggio' e 'Sistema di archiviazione'. Dopo la selezione di uno dei due elementi, il modulo selezionato correntemente viene evidenziato a destra sullo schermo. Per passare da un programma all'altro utilizzare '+', '–' o semplicemente RETURN. All'accensione verrà attivato il modulo evidenziato.

Selezionando 'Uscita' si ritorna al Menu principale.

PATTERN EDITOR

È un programma che permette di disegnare qualsiasi immagine a colori.

Quando si attiva PATTERN EDITOR, sullo schermo appare la richiesta di quale modo (mode) si desidera utilizzare. Battere il numero della modalità desiderata. Le possibilità sono:

Mode 0 (128)	–	2 colori, 640 x 256 pixel
Mode 1 (129)	–	4 colori, 320 x 256 pixel
Mode 2 (130)	–	16 colori, 160 x 256 pixel
Mode 4 (132)	–	2 colori, 320 x 256 pixel
Mode 5 (133)	–	4 colori, 160 x 256 pixel

Gli altri modi non sono adatti in quanto non possono visualizzare immagini grafiche.

Dopo la selezione dei modi, sullo schermo appare una griglia, con un cursore a croce nell'angolo superiore sinistro. Questo cursore può essere spostato sullo schermo grazie ai tasti di movimento cursore.

La gamma di colori da utilizzare per riempire ogni riquadro della griglia viene fornita nella parte inferiore dello schermo. Per riempire il riquadro contrassegnato con la croce, premere semplicemente il numero corrispondente. Ogni volta che si riempie un riquadro, il rettangolo grande nella parte superiore dello schermo viene riempito con il modello corrente. Gli otto parametri richiesti per specificare il modello corrente vengono sempre visualizzati accanto alla griglia. Notare comunque che i parametri vengono visualizzati in notazione esadecimale (preceduti da un simbolo &).

Dopo aver ottenuto il modello desiderato, prendere nota dei numeri ed utilizzarli nei vostri programmi per riempire le figure geometriche piene (quali ad esempio triangoli, cerchi o ellissi) producibili dal computer.

Sul lato destro dello schermo, è presente una striscia di blocchi colorati. Questa è la tavolozza, che permette di modificare la relazione tra i numeri di colore ed il colore corrente. Per modificare il colore, battere P.

Il puntatore vicino alla tavolozza, che compare dopo aver premuto P, può essere spostato verso l'alto o verso il basso della striscia colorata per mezzo dei tasti ↑ e ↓. Per modificare quanto prodotto da un colore premere uno dei tasti 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E o F. In questo modo si seleziona uno dei 16 colori disponibili. Per tornare alla griglia, selezionati i colori, premere **RETURN**.

Per cambiare "mode" premere **ESCAPE**.

Dispositivo/Colore

Le funzioni 'Colore Testo' e 'Colore Background' permettono di cambiare rispettivamente i colori del testo e dello sfondo, consentendo per ognuno 7 tinte diverse.

'Mouse' permette di utilizzare come opzione il Mouse anziché il Joystick o i tasti cursore.

3. Il linguaggio BASIC

Come scrivere un programma

La lingua parlata è troppo complessa per essere usata nelle comunicazioni con il computer. Tutte le istruzioni vengono invece fornite utilizzando un linguaggio per computer costituito da alcune centinaia di parole che il computer è in grado di interpretare.

Il PC 128 S è già dotato di un linguaggio flessibile e potente, il BBC BASIC, che è costituito da un certo numero di parole derivanti dall'inglese, che rendono il linguaggio stesso facile da imparare e da usare.

Nell'ultima sezione si è appreso che il PC 128 S può rispondere immediatamente ad alcuni comandi. Per esempio, se si batte:

PRINT "Salve" RETURN

il computer visualizza la parola Salve.

PRINT è una **parola chiave** BBC BASIC che il computer è in grado di riconoscere e comporta la visualizzazione di tutto ciò che segue la stessa istruzione PRINT. In questa sezione del manuale sono descritte le più importanti parole chiave BASIC e nell'Appendice 8 viene fornita una lista completa di tutte le parole chiave e del loro significato.

E' possibile che si sia già scoperto di persona l'effetto di un comando che il computer non è in grado di interpretare. Per esempio, se si batte:

PRINT "Salve RETURN

il computer darà in risposta il messaggio:

Missing " (" mancanti)

Il computer fornisce un messaggio di errore per comunicare che non è in grado di eseguire il comando impartito, in quanto non si sono seguite le regole del linguaggio BASIC. E' facile commettere errori quando si introducono istruzioni, e i messaggi di errore sono utili per la rilevazione e la correzione degli errori commessi.

Se si desidera che il computer esegua un calcolo in BASIC, si può utilizzare la normale tastiera o il tastierino numerico. Battere:

PRINT 8+7 RETURN
PRINT 20–9.5 RETURN

La moltiplicazione richiede l'uso del simbolo * e la divisione del simbolo /:

PRINT 12*9 RETURN
PRINT 25/2 RETURN

Il Basic contiene molte altre funzioni aritmetiche che possono essere utilizzate per trovare, ad esempio, la radice quadrata di un numero, o per calcolare il suo logaritmo. Provare a battere:

PRINT SQR(9) RETURN
PRINT LOG(75) RETURN

Se si ha intenzione di utilizzare il computer soprattutto per la risoluzione di calcoli complessi, ci si può servire con ottimi risultati di "VIEWSHEET".

A questo punto il contenuto dello schermo appare piuttosto disordinato, quindi battere:

CLS RETURN

che è l'istruzione BASIC per cancellare il contenuto dello schermo.

Finora si sono impartiti comandi ai quali il computer risponde immediatamente. Più spesso si fornirà al computer una serie di istruzioni numerate. Queste istruzioni verranno immagazzinate nella memoria e l'insieme delle stesse viene chiamato "programma". Il computer obbedisce alle istruzioni di programma solo quando ne viene fatta esplicita richiesta da parte dell'utente. E' possibile notare la differenza fra i due metodi battendo:

10 PRINT "Salve" RETURN

In questo caso non si ha alcun effetto e riappare il prompt >.

All'inizio della riga è stato battuto il numero 10, che è denominato numero di riga e che comunica al computer che l'istruzione successiva non deve essere eseguita immediatamente. La riga viene invece immagazzinata nella memoria, come si può vedere battendo:

LIST RETURN

Sullo schermo verrà visualizzato il programma composto da una riga. E' possibile fare in modo che il computer esegua questo breve programma battendo:

RUN RETURN

Una volta che il computer termina l'esecuzione di un programma, il prompt > ritorna sullo schermo. Ciò indica che il computer è pronto ad accettare ulteriori comandi da tastiera. Il programma rimane memorizzato, come si può verificare battendo:

LIST RETURN

Se si aggiunge un'altra riga al programma, il computer automaticamente dispone le righe in ordine numerico progressivo. Per esempio, battere:

3 PRINT "Questa è un'altra riga" RETURN

seguito da LIST per visualizzare l'intero programma.

Le righe di programma sono generalmente numerate di 10 in 10, per facilitare in seguito l'inserimento di righe addizionali. Se si batte:

RENUMBER RETURN

Il computer automaticamente rinumera il programma, assegnando alla prima riga il numero 10.

Una volta che un programma è completo, può essere memorizzato su disco in modo da poter essere usato di nuovo. Il disco Welcome contiene una serie di programmi che sono stati memorizzati in questo modo. E' probabile che non si desideri memorizzare il programma appena creato, pertanto battere:

NEW RETURN

che ordina al computer di cancellare il programma. Si può avere conferma dell'effettiva esecuzione del comando introdotto battendo il comando LIST.

Un programma può essere cancellato accidentalmente premendo o battendo NEW prima di avere memorizzato una copia del programma. Generalmente, il programma può essere recuperato a condizione che non siano state battute nuove righe di programma. Usare il comando:

OLD RETURN

per recuperare il vecchio programma.

Un semplice programma con l'utilizzo di variabili

In questa sezione e nella parte rimanente del capitolo, verrà richiesto di battere un certo numero di brevi programmi e, per maggiore chiarezza, verrà omesso il simbolo RETURN alla fine di ciascuna riga.

Battere il seguente programma:

```
10 PRINT "Puoi darmi un numero";
20 INPUT numero
30 PRINT "Hai battuto";numero
```

e quindi eseguire il programma con RUN. Il computer esegue la riga 10 e visualizza la domanda:

Puoi darmi un numero?

Il punto interrogativo viene aggiunto automaticamente in seguito all'esecuzione dell'istruzione della linea 20. L'istruzione INPUT fa in modo che il computer attenda che venga battuto qualcosa, in questo caso un numero. Battere:

6 RETURN

Una volta battuto il numero, il computer esegue la riga 30 e visualizza il messaggio:

Hai battuto 6

La riga 20 memorizza il numero in una **variabile**, così chiamata perchè il suo valore può variare. In questo caso la variabile è chiamata *numero*. Una variabile può essere paragonata ad una casella vuota che il computer riempie con un valore, in questo caso 6.

Ogni volta che il computer incontra un riferimento a numero nel programma, utilizza il valore corrente della variabile. Quindi la riga 30 visualizza "Hai battuto", seguito dal valore della variabile *numero* che è 6.

Eseguire di nuovo il programma, introducendo un numero diverso, ed osservare l'effetto. *Numero è* una variabile numerica che può essere utilizzata per immagazzinare il valore di numeri interi, decimali o negativi. Le variabili possono essere utilizzate in calcoli matematici. Aggiungere al programma le seguenti righe e rieseguire il programma:

```
40 PRINT "Due volte ";numero;" è ";2*numero
50 PRINT "Sottrai 5 da ";numero;" e ottieni ";numero-5
60 PRINT "Aggiungi 20 a ";numero;" e ottieni "; 20+numero
```

Il valore di una variabile non deve essere necessariamente introdotto, in quanto può essere fornito in modo diretto. Per esempio, battere:

```
LET altezza=2.1 RETURN
```

Quindi battere:

```
PRINT altezza RETURN
PRINT altezza*2 RETURN
```

E' possibile modificare il programma in modo da includere un'istruzione LET aggiungendo le righe seguenti:

```
35 LET numero=10
36 PRINT "Ma il nuovo valore è ";numero
```

Listare il programma in modo da visualizzare l'ordine nel quale il computer esegue le istruzioni, quindi eseguire il programma.

Nelle versioni del BASIC fornite da alcuni computer sono consentite denominazioni di variabili molto brevi, come ad esempio Q o AB. Il BBC BASIC, al contrario, consente l'uso di nomi di variabili di una certa lunghezza, cosa che rende un programma più facile da seguire e più facile da modificare. Per esempio, le seguenti istruzioni sono tutte consentite dal BBC BASIC:

```
LET Lunghezza_tappeto=7.56
LET Costo_di_3_scatole=1000
LET VELOCITA'_AUTO=60
```

(il carattere di sottolineatura si trova sullo stesso tasto del carattere £)

Sebbene tutti gli esempi precedenti utilizzino LET prima del nome di variabile, la sua inclusione non è obbligatoria. Il programma viene eseguito anche battendo:

```
35 numero=10
```

Dal momento che LET è facoltativo, si constaterà che nella maggior parte dei programmi viene omesso.

Sebbene i nomi delle variabili possano essere di qualsiasi lunghezza, devono però attenersi ad alcune semplici regole:

- La variabile deve iniziare con una lettera maiuscola o minuscola, con il segno o con il carattere di sottolineatura.
- Gli altri caratteri possono essere lettere maiuscole o minuscole, il segno o il carattere di sottolineatura, oppure dei numeri.
- Non sono consentite variabili che comincino con parole chiave Basic come PRINT o LET.

Dal momento che tutte le parole chiave Basic sono maiuscole, è facile evitare di includere parole chiave all'inizio di un nome di variabile utilizzando nella variabile solo lettere minuscole. Ciò consente inoltre ai listati di programma di essere più facilmente leggibili, visto che le variabili si distinguono nettamente.

Variabili intere

Le variabili descritte finora sono definite variabili reali, poichè possono essere utilizzate per memorizzare numeri **reali**, ovvero forniti di punto decimale. Una variabile reale può essere utilizzata per memorizzare numeri con una precisione fino a 9 cifre.

Il computer utilizza sempre la stessa quantità di memoria per memorizzare una quantità reale, anche se il numero memorizzato è un numero **intero**. Alcuni programmi richiedono solo numeri interi, e l'utilizzo di variabili reali in questo caso provoca uno spreco dello spazio disponibile in memoria e inoltre rallenta l'esecuzione del programma, poichè il computer considera 9 x 8 come 9.00000000 x 8.00000000.

Una variabile intera è un altro tipo di variabile numerica, e viene utilizzata per memorizzare esclusivamente numeri interi compresi fra –2147483648 e 2147483647. I calcoli eseguiti con variabili intere sono molto più rapidi, e le variabili in questione utilizzano un minore spazio in memoria rispetto alle variabili reali.

Una variabile intera viene sempre conclusa da un segno di percentuale, come viene mostrato nel seguente programma:

```
10 PRINT "Battere qualsiasi numero ";
20 INPUT intero%
30 PRINT "Hai battuto ";intero%
```

Eseguire il programma e introdurre, ad esempio, 4.5. Il risultato dimostra perché non si deve utilizzare una variabile intera a meno che non si sia sicuri che il valore memorizzato sia un numero intero.

Le variabili da A% a Z% sono denominate **variabili intere residenti** e, all'accensione del computer, a questo tipo di variabili viene assegnato automaticamente un determinato spazio di memoria, in modo che non debba essere utilizzato dello spazio di memoria addizionale se queste variabili vengono utilizzate in un programma.

Dopo che un programma è stato eseguito, il valore di altre variabili viene perso, ma il valore delle variabili da A% a Z% rimane immutato, anche dopo aver battuto NEW o aver premuto BREAK. Queste variabili consentono di passare informazioni da un programma all'altro.

Esiste un'altra variabile intera residente, @%. Il valore di @% è utilizzato per controllare il modo in cui il computer stampa i numeri. @% viene descritta in modo dettagliato a pagina 61.

Variabili stringa

Le variabili descritte finora sono variabili numeriche, che possono essere utilizzate soltanto per memorizzare numeri. Il computer può anche memorizzare stringhe di caratteri (cioè parole e frasi) in quelle che vengono definite **variabili stringa**. Una variabile stringa termina sempre con un segno di dollaro, come si può notare negli esempi seguenti:

```
Modello_di_Auto$="Mini Metro"

VALUTA$="Dollari"

Tempo$="Umido"
```

I caratteri fra virgolette vengono chiamati stringhe. Battere ed eseguire questo breve programma:

```
10 PRINT "Come ti chiami ";
20 INPUT nome$
30 PRINT "Piacere di conoscerti ";nome$
```

La variabile stringa nome nella riga 20 viene usata per memorizzare il nome battuto. Il contenuto di nome$ viene stampato dalla riga 30. Una variabile stringa può contenere da 0 a 255 caratteri. Si può verificare questa caratteristica eseguendo il programma alcune volte ed inserendo nomi di lunghezza diversa.

Qualsiasi set di caratteri può essere memorizzato in una variabile stringa, per esempio:

```
una_stringa_mista$="123%.abc@*"
```

Ad ogni modo, non si possono eseguire operazioni matematiche su stringhe, anche se la variabile contiene solo numeri. Così sebbene:

```
esempio$="365"
```

sia una stringa accettabile,

PRINT esempio$+5

non ha senso per il computer. Il contenuto di una variabile stringa è considerato come una serie di caratteri. Come non è possibile eseguire operazioni matematiche su quantità numeriche diverse, come ad esempio un numero civico o il numero che indica la misura di una scarpa, allo stesso modo non si possono eseguire operazioni matematiche su numeri memorizzati come stringhe.

Supporto del BBC BASIC alla programmazione

Il BBC BASIC dispone di numerose caratteristiche che rendono più semplice la programmazione. Può darsi che si siano commessi errori quando si sono battuti i programmi precedenti. In caso contrario, battere:

10 PRONT "Questo è un errore". RETURN

e verificare che cosa accade quando si esegue il programma. Il modo più lungo di correggere l'errore è di battere di nuovo l'intera riga. In alternativa si può correggere o modificare la riga utilizzando il tasto cursore e i tasti COPY posti nella parte destra della tastiera principale.

Premere ↑. Non appena si preme il tasto, il cursore si divide in due. Il cursore lampeggiante è il **cursore di copia**, che può essere spostato per copiare il testo da qualsiasi posizione dello schermo; il rettangolo bianco è il **cursore di scrittura**, che mostra dove apparirà il prossimo carattere battuto o copiato. Il cursore di scrittura si sposta solo dopo che un carattere è stato battuto o copiato.

Spostare il cursore di copia finchè non si trova al di sotto del primo carattere nella riga errata e quindi premere COPY una volta. Il numero '1' viene copiato nella posizione del carattere indicato dal cursore di scrittura. Ora premere il tasto COPY altre quattro volte per ottenere:

10 PR

Il carattere successivo non deve essere copiato, poichè è sbagliato. Battere 'I' da tastiera: quest'ultimo carattere apparirà sulla nuova riga, quindi usare → per spostare il cursore di copia al di sotto della N in "10 PRONT". E' possibile ora copiare il resto della riga per ottenere:

10 PRINT "Questo è un errore".

(Se si commettono degli errori in fase di copiatura, si può utilizzare DELETE per cancellare i caratteri battuti sulla nuova riga). Quando si è copiato l'ultimo carattere, premere RETURN. La versione corretta della riga sostituirà quella vecchia.

E' possibile spostare il cursore di copia in un'altra posizione dello schermo in qualsiasi momento in fase di copiatura, in modo da poter copiare sezioni di diverse righe differenti per creare una riga completamente nuova. Se si desidera interrompere la correzione di una riga, premere ESCAPE. **Non** premere RETURN, altrimenti la riga vecchia verrà sostituita dalla versione parzialmente corretta.

AUTO

In precedenza si è visto che le righe di programma sono generalmente numerate di 10 in 10. Ciò consente di utilizzare i numeri di riga rimasti liberi per l'inserimento successivo di eventuali istruzioni. Se lo si desidera, il computer è in grado di numerare automaticamente le righe. Trasferire il programma corrente usando NEW e quindi battere:

AUTO RETURN

Il computer visualizza 10 ed attende che si batta un'istruzione. Battere ciò che segue, ricordando di premere RETURN dopo ciascuna riga (è ancora possibile utilizzare le funzioni di editing: gran parte della riga 40 può essere copiata dalla riga 20, per esempio).

```
10 PRINT "Un programma breve"
20 PRINT "Qual'è il primo numero ";
30 INPUT primo
40 PRINT "Qual'è il secondo numero ";
50 INPUT secondo
60 PRINT primo;" più ";secondo;" fa ";primo+secondo
```

Dopo l'ultima riga il computer visualizza 70. Dal momento che il programma è completo, premere ESCAPE per impedire che il computer generi nuovi numeri di riga. E' possibile listare o eseguire il programma.

L'istruzione AUTO può essere utilizzata per cominciare la numerazione da qualsiasi numero di riga, lasciando un intervallo qualsiasi fra un numero e l'altro. L'intervallo di default è dieci. AUTO 100 produce le righe 100, 110, 120 e così via. AUTO 15,1 produce i numeri di riga 15, 16, 17, ecc.

LIST

L'istruzione LIST è già stata utilizzata ma è disponibile anche un comando LIST esteso, utile per programmi più lunghi. Provare a battere:

LIST IF PRINT RETURN

e

LIST IF primo RETURN

In altre parole, LIST IF visualizza solo quelle righe contenenti la sequenza specificata di caratteri.

DELETE

Talvolta si rende necessario eliminare delle righe da un programma. Le righe singole possono essere cancellate battendo il numero di riga e premendo RETURN. Un numero di righe in sequenza può essere cancellato utilizzando il comando DELETE. Provare a battere:

DELETE 20,50 RETURN
LIST RETURN

questa istruzione cancella i numeri di riga da 20 a 50 compresi.

RENUMBER

Se si sono inserite molte righe aggiuntive in un programma, quest'ultimo può essere riordinato utilizzando RENUMBER per distribuire i numeri di riga a intervalli di 10. La rinumerazione comincia sempre dalla prima riga del programma. Come nella funzione AUTO, si possono usare variazioni come ad esempio RENUMBER 100,5 per contrassegnare la prima riga con 100 e le successive con 105, 110, ecc.

REM

L'istruzione REM consente di introdurre delle note all'interno di un programma per ricordare la funzione di parti del programma. Sebbene nomi di variabili significativi possano ampiamente rilevare il contenuto di un programma, le REM vengono utilizzate per riassumere lo scopo di un certo numero di righe:

100 REM Righe da 110 a 150 tracciano un cerchio
500 REM Trovare il numero più grande e stamparlo

Quando un programma viene eseguito, il computer ignora qualsiasi riga che cominci con un'istruzione REM.

Abbreviazioni

Se non si è abituati all'uso di una tastiera, può sembrare un compito piuttosto fastidioso dover trovare le lettere corrette per battere, per esempio, PRINT. Il computer decifra le parole chiave BASIC se queste vengono scritte per intero o se viene utilizzata un'abbreviazione ammessa. Battere:

P."Salve" RETURN

Questo equivale esattamente a:

PRINT"Salve" RETURN

la risposta del computer è la stessa. Analogamente, I è un'abbreviazione per INPUT. Utilizzare NEW per cancellare il programma corrente, selezionare la numerazione di riga automatica e quindi battere il seguente programma, che utilizza diverse parole chiave abbreviate:

```
10 P."Scegliere un numero ";
20 I.Scelta
30 P."Numero ";scelta;"!"
40 P."Una buona scelta!"
```

Ora listare il programma. Le abbreviazioni utilizzate nelle righe di programma vengono automaticamente riportate alla loro lunghezza originale quando il programma viene listato.

Le abbreviazioni di tutte le parole chiave BASIC sono fornite nell'Appendice 8.

Uso dei tasti funzione

Gran parte dei tasti sulla tastiera visualizzano un carattere particolare quando vengono premuti. Nella parte superiore della tastiera si trova un gruppo di tasti che funzionano in modo diverso e che sono chiamati **tasti funzione**. Ciascun tasto può essere programmato per produrre un carattere o una stringa di caratteri quando viene premuto.

Per esempio, si può programmare [f0] per produrre la parola PRINT battendo:

*KEY0 PRINT [RETURN]

[f1] può essere programmato per produrre INPUT se si batte:

*KEY1 INPUT [RETURN]

Ora premere [f0] e [f1] per verificarne l'effetto. Si noterà che dopo che i caratteri sono stati visualizzati, il cursore rimane posizionato alla fine della riga. Talvolta è utile programmare un tasto funzione in modo che operi come se [RETURN] fosse stato premuto dopo la visualizzazione dei caratteri e ciò si ottiene includendo i caratteri | M nella definizione di tasto. Per esempio:

*KEY2 LIST | M [RETURN]

provoca il listato del programma corrente ogni volta che viene premuto [f2]. Alcune modalità di schermo consentono di visualizzare solo 20 caratteri per riga, cosa che rende piuttosto difficoltosa la lettura di un listato. Pertanto sarebbe probabilmente meglio definire [f2] in modo che il computer passi alla modalità 135, che è la più leggibile fra quelle disponibili, prima di eseguire il listato di un programma:

*KEY2 MODE135 | MLIST | M [RETURN]

E' utile scrivere un breve programma che definisca i tasti. Questo programma può essere caricato ed eseguito all'inizio di una sessione al computer. Le definizioni di tasto conservano la loro impostazione finchè:

– i tasti sono ridefiniti;

– viene utilizzata un'istruzione *FX18, che annulla l'impostazione dei tasti;

– si esegue un hard break (con la pressione di CTRL + BREAK).

Battere il seguente programma, che imposta tutti i tasti funzione:

```
10*KEY0 MODE135 | MLIST | M
20*KEY1 RUN | M
30*KEY2 MODE
40*KEY3 PRINT
50*KEY4 INPUT
60*KEY5 COLOUR
70*KEY6 MOVE
80*KEY7 DRAW
90*KEY8 PLOT
100*KEY9 GCOL
```

In seguito è possibile che si desideri utilizzare le definizioni di tasto per conto proprio, ma ci si accorgerà che il programma precedente è comunque utile per i prossimi capitoli. La sezione seguente mostra come si può memorizzare il programma appena creato, in modo che sia disponibile in seguito.

Memorizzazione e caricamento di un programma

La maggior parte dei programmi che si sono appena battuti hanno una lunghezza abbastanza limitata e non hanno una funzione di particolare rilievo. Pertanto, non è consigliabile conservarne una copia permanente su disco ma, man mano che si accresce la propria conoscenza della programmazione in BASIC, è probabile che si desiderino conservare versioni di programma scritte in modo che possano essere eseguite senza dover battere di nuovo tutte le istruzioni.

L'esecuzione di una copia permanente di un programma è definita **salvataggio** e il linguaggio BASIC fornisce a questo scopo un comando particolare. Il suo formato è:

SAVE "nome" RETURN

dove "*nome*" (che deve essere racchiuso fra virgolette) serve per contraddistinguere il programma in questione da tutti gli altri.

Nota: Se si desidera eseguire i comandi forniti qui sotto, è necessario un disco formattato (**non** il disco Welcome). Per informazioni su come produrre un disco formattato consultare la sezione "Sistemi di archiviazione".

Pertanto, per salvare il programma di definizione dei tasti funzione che si è appena introdotto, battere:

SAVE "TASTI" RETURN

o eventualmente:

SAVE "DEFTASTI" RETURN

o un comando SAVE che includa un nome a scelta.

Introdurre il disco nel drive e battere:

*MOUNT 0

seguito dal comando SAVE scelto (ricordare di non utilizzare il disco Welcome).

Non appena si preme RETURN, la spia luminosa del drive si accende e il motorino emette un ronzio durante il salvataggio.

Il prompt > appare nuovamente quando il programma è stato salvato.

Si noti che SAVE si limita a trasmettere una copia del programma, che in ogni caso rimane in memoria per l'esecuzione, la visualizzazione e eventuali modifiche.

Il processo di recupero di un programma da disco è definito caricamento e, anche in questo caso, il linguaggio BASIC fornisce un comando particolare:

LOAD "nome" RETURN

Ovviamente, deve esistere un programma con tale nome perchè possa essere caricato.

Utilizzando NEW cancellare dalla memoria il programma di definizione dei tasti funzione.

Battere il comando LOAD seguito dal nome del programma di definizione dei tasti funzione e premere RETURN. Il prompt > apparirà di nuovo non appena il programma è caricato.

Listare il programma per verificare che sia stato caricato.

Notare che l'operazione di caricamento sostituisce il programma corrente, pertanto, se non lo si vuole perdere, occorre accertarsi di averlo salvato.

Programmare o non programmare

Nelle sezioni precedenti è stata fornita una breve descrizione di alcune delle funzioni di programmazione in BASIC per il computer PC 128 S. Nel caso si desideri saperne di più, consultare le sezioni successive.

E' possibile che, invece, si ritenga di avere già sufficienti nozioni riguardo alla programmazione. Ci si può chiedere, ad esempio, se siano davvero necessarie così tante nozioni prima di poter utilizzare il computer di cui si dispone.

A questo punto vale la pena ricordare il modo in cui il computer possa essere utilizzato, a seconda delle diverse categorie di utenti.

Migliaia di persone si dilettano a creare propri programmi di gioco oppure a gestire il bilancio mensile, fanno parte di computer club e scambiano consigli e notizie con altri appassionati.

Altri invece non vanno oltre i principi fondamentali di programmazione. Quando queste persone desiderano catalogare la propria collezione di francobolli, semplicemente acquistano un programma già pronto direttamente in un negozio o per corrispondenza. Piuttosto che affannarsi a scrivere un semplice programma preferiscono acquistare software molto complicato creato dopo mesi di fatica da parte di programmatori professionisti.

Alcuni utenti utilizzano occasionalmente il computer per giocare, ma il loro scopo principale è un'applicazione professionale per preparare e stampare lettere con un word processor o elaboratore testi, utilizzare uno spreadsheet o foglio elettronico come aiuto per prendere decisioni finanziarie e memorizzare dati importanti su disco.

Il computer è uno strumento, complesso e sofisticato, ma pur sempre uno strumento. Per poter lavorare con un computer non è assolutamente necessario saper programmare. Il computer mette a disposizione dell'utente potenti funzioni, che verranno descritte più avanti nel manuale. Per sfruttare le capacità di un computer non è necessario saper programmare, in quanto sul mercato sono disponibili programmi per ogni esigenza.

Nei prossimi capitoli verranno descritte alcune delle possibilità di programmazione. Queste informazioni vengono fornite quale introduzione al BASIC, ma possono essere comprese a fondo eseguendo gli esempi di programma forniti nel corso del manuale. Per aggiungere nuove righe modificare semplicemente i valori delle variabili. Se si commette un errore che provoca l'esecuzione all'infinito del programma, non c'è da preoccuparsi. Premere ESCAPE per arrestare il programma e ritornare al prompt > che indica che a questo punto possono essere introdotte istruzioni.

Grafica

Il PC 128 S offre otto diversi modi di visualizzazione ed i programmi "Tutorials", contenuti nel software Welcome, dimostrano come il numero di caratteri che compongono una riga può variare a seconda della modalità. Questi programmi ci mostrano inoltre come alcune modalità possano sia stampare testo sia visualizzare figure.

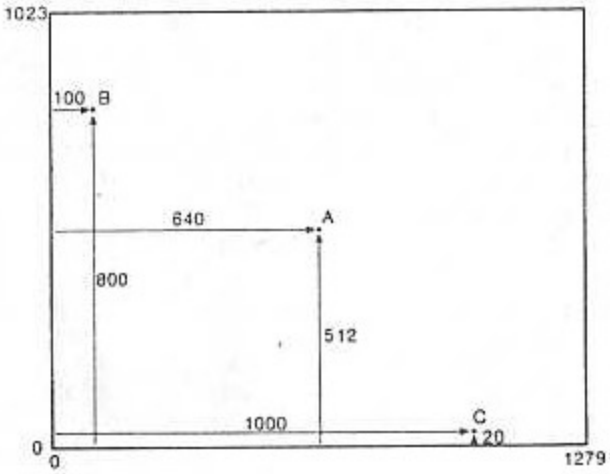
I diversi modi differiscono considerevolmente gli uni dagl'altri. Alcune di queste differenze verranno spiegate nel presente capitolo, mentre la lista completa delle caratteristiche di ciascuna modalità viene fornita nell'Appendice 2.

Sono disponibili due insiemi di modi: i modi da 0 a 7 e i modi da 128 a 135. Ciascun modo N del primo insieme ha un modo corrispondente N+128 nel secondo insieme che funziona apparentemente allo stesso modo e che possiede le stesse caratteristiche. Ad esempio, il modo 7 e il modo 135 hanno un aspetto praticamente identico.

Nei PC 128 S i modi da 128 a 135 offrono maggiore spazio di memoria per scrivere programmi estesi. Per questo motivo si dovrebbero sempre utilizzare i modi del secondo insieme ma i programmi di esempio che seguono e che utilizzano il secondo insieme di modi possono essere eseguiti senza problemi con i corrispondenti modi del primo insieme.

I modi 128, 129, 130, 132 e 133 sono definiti **modi grafici** in quanto possono gestire sia testo sia immagini. I modi 131 e 134 sono modi di solo testo. Il modo 135 permette di utilizzare funzioni grafiche, ma i comandi coinvolti sono molto diversi e quindi verranno descritti in un capitolo a parte.

Nei modi grafici i punti sullo schermo sono espressi sotto forma di coordinate, in modo che la loro posizione possa essere facilmente identificata.



Le coordinate del punto A nella figura sono: 640 orizzontale e 512 verticale, praticamente il centro dello schermo. Il punto B è posizionato alle coordinate 100,800, ed il punto C alle coordinate 1000,20.

Battere ed eseguire il programma seguente:

```
10 MODE 128
20 MOVE 100,100
30 DRAW 800,100
40 DRAW 800,900
50 DRAW 100,100
```

La riga 10 attiva il modo grafico. Il cursore invisibile viene posizionato automaticamente alle coordinate 0,0, cioè nell'angolo inferiore sinistro dello schermo grafico.

La riga 20 provoca lo spostamento alla posizione 100,100, senza tracciare una linea.

Il comando DRAW traccia una linea a partire dall'ultimo punto (100,100) fino a 800,100. I comandi DRAW rimanenti creano una serie di linee che formano un triangolo.

Precedentemente si era visto che, dopo aver eseguito il programma, lo schermo può essere cancellato digitando:

CLS RETURN

Lo schermo può essere pulito anche introducendo:

CLG RETURN

Sebbene possa sembrare che entrambi i comandi producano lo stesso effetto, CLS serve a cancellare lo schermo di testo, mentre CLG cancella lo schermo grafico. Generalmente sia il testo sia la grafica possono trovarsi in qualsiasi punto dello schermo. Più avanti vedremo come l'area di testo e l'area grafica possano essere visualizzate separate, e a quel punto si presenterà la necessità di avere due comandi separati per cancellare lo schermo.

Le linee tracciate nel modo 128 sono le linee più sottili che possono essere prodotte con il computer. Per questo motivo questo modo si dimostrerà particolarmente utile per la creazione di immagini molto accurate ad **alta risoluzione**. Lo stesso programma può essere eseguito in altri modi grafici. A dimostrazione di questo, modificare la riga 10 come mostrato di seguito ed eseguire nuovamente il programma. Battere:

10 MODE 129 RETURN
RUN RETURN

Con questo modo le linee tracciate saranno più larghe in quanto il modo 129 è un modo a media risoluzione. Il vantaggio maggiore di quest'ultimo modo rispetto alla 128 è che permette di visualizzare quattro colori contemporaneamente. Il colore delle linee può essere modificato digitando:

35 GCOL 0,1
45 GCOL 0,2

ed eseguendo nuovamente il programma. GCOL può essere utilizzato per selezionare il colore da utilizzare in un'istruzione DRAW. Il numero che segue GCOL (0), si riferisce ad un particolare colore a seconda del modo. Nel modo 129:

GCOL 0,0 crea linee nere
GCOL 0,1 crea linee rosse
GCOL 0,2 crea linee gialle
GCOL 0,3 crea linee bianche

Dopo che un colore è stato selezionato, rimarrà attivo per tutte le istruzioni DRAW finché non viene utilizzato un nuovo comando GCOL.

GCOL può inoltre essere utilizzato per modificare il colore di sfondo dell'immagine. Ad esempio, battendo:

```
MODE 129 RETURN
GCOL 0,130 RETURN
CLG RETURN
```

si imposterà il colore di sfondo su giallo, cioè l'intero schermo diventa giallo. Tutti i numeri GCOL maggiori di 127 modificano il colore di sfondo.

Eseguire nuovamente il programma dopo aver modificato la riga 10 come segue:

```
10 MODE 130
```

Il modo 130 è un modo a bassa risoluzione che crea linee molto larghe e visualizza 16 colori simultaneamente sullo schermo, otto dei quali sono colori lampeggianti.

Notare che GCOL 0,2 in questo modo imposta lo schermo su verde e non giallo. I riferimenti numerici di colore non sono sempre gli stessi per tutti i modi grafici. Per un dettaglio dei colori disponibili con GCOL consultare l'Appendice 2.

Nel modo 130, GCOL 0 può essere seguito da un qualsiasi numero compreso nella gamma da 0 a 15 per la selezione di un colore. Modificare le istruzioni GCOL ed osservare gli effetti delle modifiche apportate.

Il comando PLOT

Il comando PLOT è un comando multi–uso. I comandi MOVE e DRAW costituiscono esempi particolari di PLOT. I comandi MOVE e DRAW vengono utilizzati molto frequentemente, per questo motivo anche ai comandi PLOT che producono gli stessi effetti sono state assegnate le stesse parole chiave:

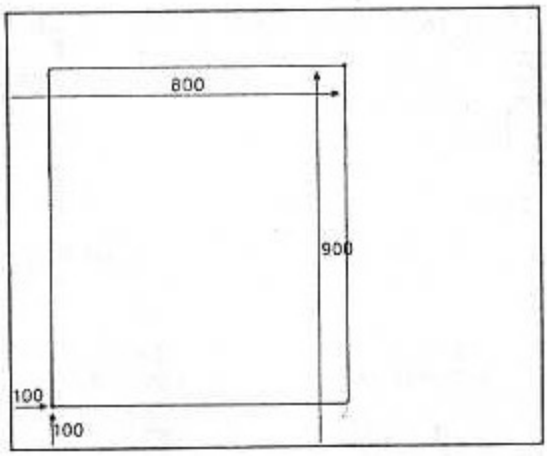
PLOT 4,100,100 equivale a MOVE 100,100

PLOT 5,800,100 equivale a DRAW 800,100

Il primo numero dopo PLOT determina il modo in cui le linee verranno tracciate. Con i comandi PLOT si possono creare rettangoli, parallelogrammi, cerchi, segmenti, settori, archi, triangoli o ellissi, di cui si può visualizzare o il contorno, o la figura piena di colore o di un modello specifico. Nei programmi Welcome si era potuto avere una dimostrazione dei diversi tipi di visualizzazione. Per tracciare un rettangolo pieno, digitare:

```
10 MODE 129
15 REM spostamento su un angolo del rettangolo
20 MOVE 100,100
25 REM spostamento sull'angolo diagonalmente opposto
30 PLOT 101,800,900
```

PLOT 101 ordina al computer di tracciare un rettangolo i cui angoli opposti sono rappresentati dall'ultimo punto su cui ci si era posizionati ed il punto presente:

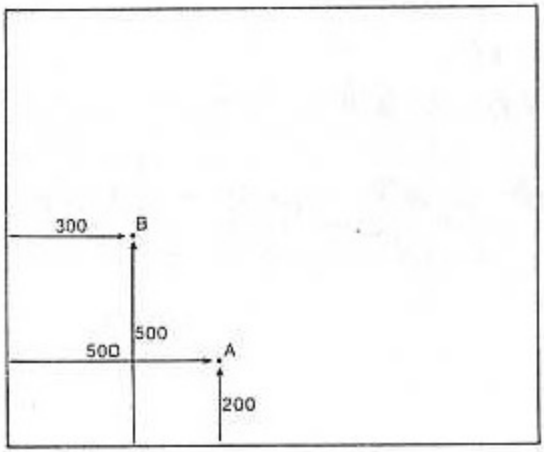


Modificare la riga 20 come mostrato di seguito e rieseguire (RUN) il programma:

20 MOVE 300,100

Come si può notare, il rettangolo è stato rimpicciolito in quanto la posizione del primo angolo è stata modificata. Per tracciare una serie di rettangoli identici in diverse posizioni, occorre calcolare la nuova posizione dell'angolo opposto di ciascun rettangolo. Per evitare questa lunga procedura, è disponibile un altro comando PLOT che permette di impostare il rettangolo in modo diverso.

Invece di indicare una posizione esatta o assoluta di un punto sullo schermo, è possibile indicare la sua distanza da un altro punto. Questa viene chiamata posizione relativa del punto:



Il punto A nella figura è posizionato sulle coordinate 500,200. Il punto B è impostato 200 punti a sinistra e 300 punti sopra il punto A, quindi la sua posizione relativa sarà –200,300. Il programma che segue traccia un rettangolo utilizzando A e B quali posizioni dei due angoli:

```
10 MODE 129
20 MOVE 500,200
30 PLOT 97,–200,300
```

Modificare ora la riga 20 come segue e rieseguire il programma:

```
20 MOVE 300,100
```

PLOT 97 ordina al computer di tracciare un rettangolo per mezzo dei due punti introdotti, con il secondo punto relativo al primo punto. Ciò significa che il computer traccerà sempre un rettangolo di uguale misura, indipendentemente da dove sia posizionato il primo punto. Il posizionamento relativo risulta molto utile per spostare un'immagine sullo schermo.

I comandi PLOT sono molto versatili e permettono di avere il controllo sul tracciamento delle immagini sullo schermo. Le linee o le figure possono essere tracciate in modo assoluto o relativo, piene o tratteggiate, nel colore di primo piano o di sfondo. Le figure possono essere vuote o piene di colore. L'Appendice 9 fornisce una lista completa dei comandi PLOT.

Per tracciare un cerchio è necessario introdurre la posizione del centro e un punto della circonferenza:

```
10 MODE 1
15 REM coordinata del centro
20 MOVE 300,300
25 REM coordinate del punto sulla circonferenza
30 PLOT 149,550,300
```

Di seguito viene fornito un esempio di comando PLOT per tracciare una figura piena. Modificare la riga 30 come segue:

```
30 PLOT 157,550,300
```

e rieseguire il programma. Per ottenere un cerchio rosso, digitare:

```
16 GCOL 0,1
```

Altri comandi PLOT permettono di creare altre figure piene (rettangoli, ellissi, settori di un cerchio, ecc.). Si potranno creare figure più complesse partendo da queste semplici figure. Dopo il tracciamento, tutte le figure potranno essere riempite di colore.

```
1 REM faccia – un capolavoro incompiuto
10 MODE 130
19 REM selezione rosso
20 GCOL 0,1
29 REM disegno testa
30 MOVE 500,500
40 PLOT 149,800,500
49 REM occhio destro
50 MOVE 620,600
60 PLOT 149,680,600
69 REM occhio sinistro
70 MOVE 380,600
80 PLOT 149,440,600
89 REM naso rettangolare
90 MOVE 460,600
100 PLOT 101,540,400
109 REM arco per rappresentare la bocca
110 MOVE 500,600
120 MOVE 350,350
130 PLOT 165,650,350
139 REM modifica colore in giallo
140 GCOL 0,3
150 PLOT 133,500,320
```

Completare il disegno aggiungendo orecchie e colorando gli occhi.

Il programma "faccia" funziona in modo 130, il modo che permette di utilizzare 16 colori diversi. Negli altri modi, quali ad esempio il 129, possono essere visualizzati solo quattro colori contemporaneamente, La gamma dei colori viene aumentata di quattro tinte supplementari composte da una combinazione di diversi colori. Ad esempio, nel modo 129:

GCOL 16,0 rosso–arancio
GCOL 32,0 arancio
GCOL 48,0 giallo–arancio
GCOL 64,0 panna

Questi colori vengono prodotti indipendentemente dal secondo numero utilizzato. L'effetto dei comandi varia a seconda del modo in quanto i colori supplementari vengono creati in base ai colori disponibili in ciascun modo. Provare a modificare i comandi GCOL nel programma "faccia" precedente per produrre alcuni dei modelli disponibili nel modo 130.

Per creare propri colori consultare la sezione a pag. 96.

Stampa del testo

Il testo può essere visualizzato in una qualsiasi degli otto modi disponibili, ma a seconda del modo utilizzato varia il numero dei caratteri per riga (20, 40 o 80). Provare a digitare quanto segue:

```
10 MODE 128
20 PRINT "Questa è una frase"
30 PRINT "stampata."
```

Eseguire il programma alcune volte dopo aver modificato 128 nella riga 10 in: 129, 130 o 135. Il modo 135 offre una visualizzazione più chiara. Se si utilizza un televisore al posto di un monitor, il testo prodotto in modo 128 potrebbe risultare quasi illeggibile.

Dopo aver eseguito un'istruzione PRINT il computer si sposta sull'inizio della riga successiva salvo indicazioni contrarie. Eseguire nuovamente il programma dopo aver modificato la riga 20 come segue:

```
20 PRINT "Questa è una frase";
```

Il simbolo di punto e virgola alla fine della riga serve a comunicare al computer di rimanere sulla stessa riga dopo aver stampato la stringa. Il risultato sarà il seguente:

```
Questa è una frasestampata.
```

Il punto e virgola è molto utile quando si stampa una variabile all'interno di una frase e si desidera che tutto il testo rimanga sulla stessa riga. Aggiungere le seguenti righe al programma:

```
40 età=105
50 PRINT "Io ho ";età;" anni."
```

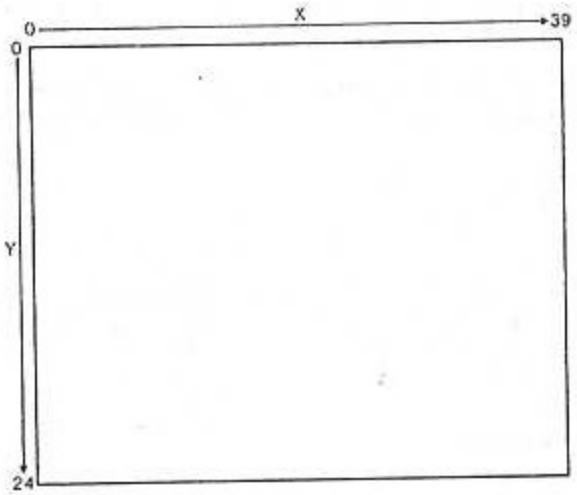
Gli spazi all'interno della riga 50 sono molto importanti in quanto evitano che le parole vengano stampate una attaccata all'altra, come avviene nel primo esempio.

Includendo degli apostrofi, si provoca la stampa di righe vuote. Ad esempio:

```
50 PRINT ""Io ho ";età;" anni."
```

stamperà due righe vuote prima della riga effettiva di output.

La posizione dei caratteri sullo schermo può essere descritta in termini di *coordinate di testo*. Le coordinate di testo, contrariamente alle coordinate per una figura, vengono calcolate in base all'angolo superiore sinistro dello schermo. Nel modo 135 le coordinate di testo sono:



Notare che sebbene su una riga vi sia spazio per 40 caratteri, le posizioni di carattere vengono calcolate da 0 a 39. Ricordare che anche le righe vengono calcolate a partire dalla numero 0.

L'istruzione PRINT TAB permette di controllare il punto da cui partirà la visualizzazione. Utilizzare NEW per cancellare il programma corrente e battere:

```
10 MODE 135
20 PRINT "0123456789"
30 PRINT TAB(5);"Un esempio di tabulazione."
```

All'esecuzione, il risultato sarà:

```
0123456789

     Un esempio di tabulazione
```

La stampa inizia dalla posizione cinque della riga, cioè dalla sesta colonna. Sulla stessa riga possono essere utilizzate più posizioni di tabulazione; quando però il computer ha già superato una posizione di tabulazione si verrà riportati all'inizio della riga successiva. Ad esempio:

```
30 PRINT
TAB(5);"Un";TAB(10);"esempio";TAB(15);"di";TAB(20);"tabulazione."
```

darà come risultato:

```
0123456789

     Un esempio
                di tabulazione.
```

Il computer si trova già alla posizione 17 quando incontra il comando TAB(15), quindi riprende sulla riga successiva.

Inserendo il numero di riga, è possibile utilizzare il comando PRINT TAB per inserire del testo in qualsiasi punto sullo schermo. Ad esempio:

```
10 MODE 135
20 PRINT TAB(8,24) "Può spostarsi in basso"
30 PRINT TAB(14,0) "O in alto"
40 PRINT TAB(1,11) "O a sinistra";TAB(27);"O a destra"
```

La riga 30 dovrebbe ricordare che sebbene il modo 135 disponga di 25 righe, queste sono numerate da zero a 24. La riga 40 mostra come sia possibile utilizzare TAB all'interno di una riga senza far riferimento ad un numero di riga.

Stampa a colori del testo

Il comando COLOUR permette di modificare i colori utilizzati per la visualizzazione del testo. Battere:

```
MODE 129 RETURN
COLOUR 1 RETURN
```

Il numero dopo COLOUR imposta il colore rosso in modo 129 e comunica al computer che il colore di primo piano del nuovo testo dovrà essere rosso. Tutto quello che verrà introdotto da questo punto in poi sarà di colore rosso. Digitare:

```
COLOUR 2 RETURN
COLOUR 129 RETURN
```

Il primo comando COLOUR modifica il colore del testo in giallo, mentre il secondo imposta il colore di sfondo su rosso. Tutto il testo introdotto da questo punto in poi verrà stampato giallo su sfondo rosso. Per impostare l'intero schermo sul nuovo colore di sfondo, battere:

```
CLS RETURN
```

I comandi COLOUR possono essere utilizzati in tutti i modi ad eccezione del modo 7 e 135. Come avviene per GCOL, con COLOUR i numeri utilizzati per indicare un particolare colore variano a seconda del modo. La lista completa dei riferimenti di colore per ogni modo è riportata nell'Appendice 2.

Formattazione più sofisticata di stampa

Per creare una tabella di cifre è possibile visualizzare i numeri su posizioni specifiche senza dover utilizzare ogni volta il comando TAB introducendo delle virgole di separazione tra un valore e l'altro.

```
10 MODE 135
15 REM le righe 20 e 30
10 REM mostrano le posizioni dei caratteri
20 PRINT TAB(10) "1111111111222222222223"
30 PRINT "01234567890123456789012345678 90"
40 PRINT 1.23,4.567,89
```

Il risultato sarà:

```
          1111111111222222222223
01234567890123456789012345678 90
      1. 23      4. 567          89
```

Ciascun numero viene visualizzato a destra di una colonna larga 10 caratteri. Queste colonne sono chiamate campi e la larghezza di ciascun campo è impostata a 10 caratteri all'accensione del computer.

Il testo viene stampato a sinistra del campo, come si può notare battendo quanto segue:

```
50 PRINT "Ciao","caro","amico"
```

che produrrà:

```
          1111111111222222222223
01234567890123456789012345678 90
      1. 23      4. 567          89
Ciao        caro        amico
```

Quando un numero o una parola sono più lunghi della larghezza del campo, l'elemento successivo viene stampato nel prossimo campo vuoto. Ad esempio:

```
          1111111111222222222223
01234567890123456789012345678 90
      1. 23      4. 567          89
Complimenti             caro        amico
```

Il campo può avere una larghezza compresa tra 0 e 255 caratteri. Modificare la larghezza del campo come segue:

```
10 MODE 135
18 REM impostazione della larghezza del campo a 8 caratteri 19 REM creando 5
campi sullo schermo
20 @%=&08
30 PRINT TAB(8) "Profitti"
40 PRINT '"gen",1234.56,789,123.45,678.9
50 PRINT "feb",234.5,67.89,12,3456.78
```

@% nella riga 20 è una variabile speciale che controlla la stampa di numeri. In questo caso viene utilizzata solo per ridurre la larghezza di campo a otto caratteri.

Il risultato è molto disordinato e confuso in quanto i numeri non sono allineati verticalmente in base al punto decimale. Modificare la riga 20 come segue:

```
20 @%=&020208
```

In questo modo si comunica al computer di stampare ogni numero con due posizioni decimali e con un campo di otto caratteri.

@% permette di avere il massimo controllo sul modo in cui il computer visualizza i numeri. Questa istruzione viene descritta in dettaglio nei manuali di riferimento. Notare che, dopo aver impostato @%, questa istruzione rimane attiva finchè non viene ripristinata, si esegue un'interruzione o si spegne il computer.

Testo e grafica

Può a volte rendersi utile mantenere la visualizzazione del testo in una porzione particolare di schermo, come si è visto nel programma Welcome. Il programma Tartaruga (TURTLE) permette di introdurre i comandi da tastiera, ma questi comandi vengono visualizzati nelle ultime quattro righe dello schermo, in modo da non disturbare la visualizzazione.

Battere:

MODE 135 RETURN

Per creare una finestra di testo all'interno del testo visualizzato, battere:

VDU 28,12,15,30,10 RETURN

Battere alcuni caratteri da tastiera. Il testo verrà visualizzato all'interno della finestra al centro dello schermo.

Il comando VDU 28 fa parte di una serie di comandi VDU che permettono di controllare la visualizzazione di testo e immagini sullo schermo. I comandi VDU possono essere utilizzati per modificare i colori di testo e immagini, per spostare il cursore, cancellare l'intero schermo, ecc. VDU 28 si utilizza specificatamente per impostare una finestra di testo.

I primi due numeri che seguono VDU 28 forniscono la posizione dell'ultimo carattere in basso a sinistra nella finestra di testo. I rimanenti due numeri indicano la posizione del carattere in alto a destra nella finestra (vedere figura).

Dopo aver creato una finestra di testo, la posizione in alto a sinistra all'interno della finestra sarà 0,0. Tutti i comandi PRINT TAB si riferiranno a questa nuova posizione, come dimostrato dal risultato delle righe seguenti:

CLS RETURN
PRINT TAB(4,3)"Centro" RETURN

In questo caso vengono utilizzate le coordinate grafiche. Notare che i numeri sono separati da punti e virgola, al contrario di quanto avviene per il comando VDU 28.

Per visualizzare la finestra grafica, battere:

GCOL 0,130 RETURN
CLG RETURN

Sebbene le figure vengano ora visualizzate esclusivamente all'interno della finestra, lo schermo completo viene utilizzato per il testo. Per separare completamente il testo dalle figure, occorrerà impostare due finestre separate. Per creare una finestra di testo al di sotto della finestra grafica, digitare:

VDU 28,5,31,34,28 RETURN

Impostare il colore di sfondo del testo su rosso, come segue:

COLOUR 129 RETURN
CLS RETURN

Introdurre alcuni comandi MOVE e DRAW. Il testo viene visualizzato all'interno della finestra di testo e tutte le linee tracciate vengono visualizzate esclusivamente all'interno della finestra grafica.

Dopo un comando VDU 28 lo schermo viene utilizzato ancora per testo e figure. Quindi, per riportare la visualizzazione allo stato normale, digitare:

VDU 26

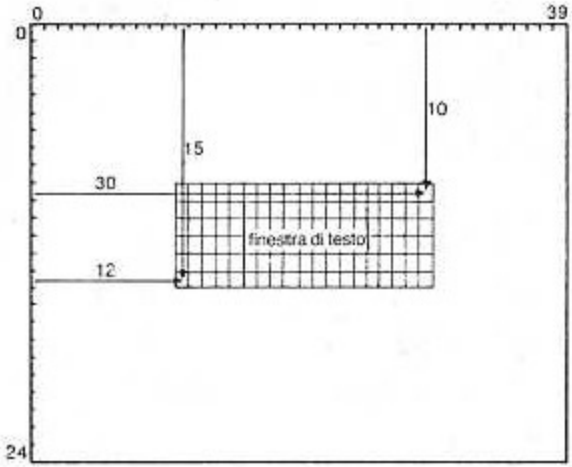
Notare che lo stesso effetto si ottiene con un comando MODE, che cancella automaticamente le finestre dallo schermo.

Stampa di testo alle posizioni grafiche

TAB viene utilizzato per visualizzare i caratteri alle coordinate di testo. A volte può rendersi utile posizionare il testo in modo più preciso sullo schermo di quanto può essere ottenuto con il comando PRINT TAB, in special modo se devono essere introdotte anche figure.

Nei modi che supportano funzioni grafiche, il testo può essere visualizzato alle posizioni grafiche per mezzo del comando VDU 5. Il programma che segue crea un effetto tridimensionale stampando lo stesso messaggio due volte e sfalsando leggermente il secondo set di caratteri che viene stampato con un altro colore:

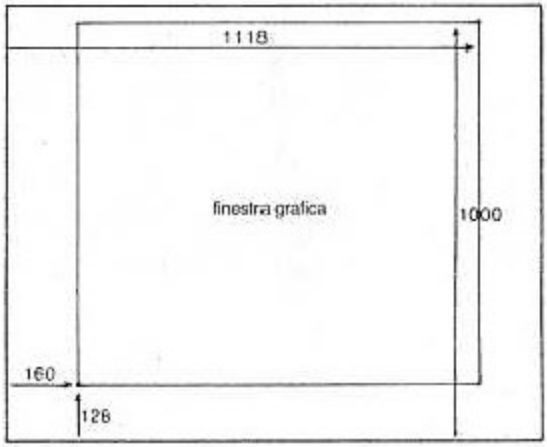
```
10 MODE 1
20 PRINTTAB(16,15)"C i a o"
30 VDU 5
40 GCOL 0,1
50 MOVE 516,540
60 PRINT"C i a o"
70 VDU 4
```



Le finestre per le immagini vengono impostate in tutte le modalità che supportano le funzioni grafiche. Battere:

MODE 129 RETURN
VDU 24,160;128;1118;1000; RETURN

VDU 24 è seguito dalle posizioni grafiche (in basso a sinistra e in alto a destra nella nuova finestra):



La riga 20 stampa i caratteri utilizzando le normali coordinate di testo. VDU 5 nella riga 30 unisce il cursore per il testo ed il cursore grafico. Nella riga 50 l'istruzione MOVE posiziona il testo, che viene stampato alle coordinate grafiche. Infine, il comando VDU 4 riporta il cursore di testo alla condizione normale in modo da poter utilizzare nuovamente l'istruzione PRINT TAB.

Input

Precedentemente si era visto come battere informazioni da tastiera in fase di esecuzione del programma, grazie all'istruzione INPUT:

```
10 MODE 135
20 PRINT "Quanti anni hai";
30 INPUT età
40 PRINT "Quindi hai ";età;" anni."
50 PRINT "Non li dimostri"
```

L'istruzione INPUT nella riga 30 provoca la visualizzazione di un punto interrogativo. Il computer attende l'introduzione di una risposta da parte dell'utente. La risposta deve essere sotto forma di numero, in quanto *età* è una variabile numerica. Dopo aver premuto RETURN, il computer memorizza il valore introdotto dall'utente per *età*. Introducendo un carattere alfabetico, il computer assume il valore zero.

Per introdurre del testo, occorre utilizzare una variabile stringa nell'istruzione INPUT:

```
10 MODE 135
20 PRINT "Come ti chiami";
30 INPUT nome$
40 PRINT "Ciao ";nome$;". Come stai?"
```

E' possibile utilizzare una sola istruzione INPUT per porre più domande:

```
10 MODE 135
20 PRINT "Come ti chiami e quanti anni hai ";
30 INPUT nome$, età
40 PRINT "Ciao ";nome$;". E così hai ";età;" anni."
```

In questo caso il computer attenderà due introduzioni da tastiera, una variabile stringa ed una variabile numerica. Le risposte possono essere introdotte separandole da una virgola, oppure possono essere entrambe seguite da un RETURN.

L'istruzione PRINT prima dell'istruzione INPUT fornisce un messaggio che ricorda cosa deve essere introdotto. Questo messaggio può essere compreso nell'istruzione INPUT:

```
10 MODE 135
20 INPUT "Come ti chiami ";nome$
30 PRINT "Ciao ";nome$;". Piacere di conoscerti."
```

INPUT ignora gli spazi all'inizio di un'introduzione e tutto quanto si trova dopo una virgola:

```
Come ti chiami ? Nerone, Imperatore di Roma
Ciao Nerone. Piacere di conoscerti.
```

Per introdurre un testo che comprenda degli spazi iniziali o che contenga delle virgole, sarà necessario utilizzare INPUT LINE al posto di INPUT:

```
20 INPUT LINE "Come ti chiami ";nome$
```

Il risultato sarà:

```
Come ti chiami ? Nerone, Imperatore di Roma
Ciao Nerone, Imperatore di Roma. Piacere di conoscerti.
```

GET e INKEY

In alcuni programmi, come ad esempio nei giochi, il computer deve reagire subito dopo la pressione di un tasto. Tali programmi utilizzano GET o INKEY al posto delle istruzioni INPUT. GET attende la pressione di un tasto prima di proseguire:

```
10 MODE 135
20 PRINT "Premere un tasto per continuare"
30 scelta=GET
40 PRINT "Il programma è terminato."
```

Con l'istruzione INKEY il computer attende la pressione di un tasto entro un tempo prefissato:

```
10 MODE 135
20 PRINT "Premere un tasto per continuare"
30 PRINT "Hai a disposizione solo 3 secondi"
40 scelta=INKEY(300)
50 PRINT "Il programma è terminato."
```

Il valore per il tempo è calcolato in centesimi di secondo, quindi con la riga 40 il computer attende per tre secondi la pressione di un tasto (300 centesimi di secondo). Se entro tale termine non viene premuto alcun tasto, il computer prosegue con la riga di programma successiva. Premendo un tasto il computer passa immediatamente alla riga successiva.

Codici ASCII

Sia l'istruzione GET sia INKEY producono quello che viene chiamato codice ASCII della pressione di un tasto. Internamente, il computer utilizza un numero da 0 a 255 per rappresentare ogni carattere contenuto nella memoria. Questo numero è il codice ASCII del carattere. Ad esempio, il codice ASCII di A è 65, di B è 66 e di C è 67. Quindi il computer memorizzerà ABC come 65, 66 e 67.

Il computer può fornire il codice ASCII di un carattere. Ad esempio:

PRINT ASC"A" RETURN

stamperà 65. Notare che ASC funziona con un solo carattere. Per avere i codici ASCII di più caratteri, consultare la tabella nell'Appendice 3 che fornisce il set completo di caratteri.

Nei due programmi precedenti il codice ASCII veniva memorizzato nella variabile *scelta*. Non premendo alcun tasto entro il termine prefissato, a *scelta* viene assegnato il valore –1.

GET o INKEY non visualizzano automaticamente il carattere battuto da tastiera. Questa funzione è utile quando non si desidera alcuna visualizzazione. Per stampare il carattere, utilizzare PRINT CHR$ per convertire il codice ASCII in stringa:

```
10 MODE 135
20 PRINT "Battere un carattere – ";
30 scelta=GET
40 PRINT CHR$scelta
50 PRINT "Hai battuto ";CHR$scelta
```

VDU seguito da un codice ASCII avrà lo stesso effetto di PRINT CHR$. Infatti PRINT CHR$(65) e VDU 65 stamperanno entrambi la lettera A. Introducendo:

VDU 66,66,67 RETURN

il computer visualizzerà quanto segue:

BBC

I codici ASCII per i caratteri partono dal numero 32. I codici con valore inferiore vengono utilizzati per impartire comandi al computer, come si è visto con VDU 26 e VDU 28.

Programmi strutturati

Nell'ultima sezione di questo manuale si erano visti i comandi più usati dal BBC BASIC. La maggior parte di questi comandi ha a che vedere con il modo in cui si comunica con il computer in fase di esecuzione di un programma e con il modo in cui è possibile modificare la visualizzazione.

Come si è visto, i programmi sono molto più leggibili quando contengono nomi di variabili sensati. Inoltre, tutti gli esempi di programma utilizzati fino ad ora contenevano una sola istruzione per riga. I programmi possono però contenere più istruzioni per riga, purchè siano separate da due punti:

```
10 MODE 135:PRINT "Battere un carattere – ";:scelta=GET: PRINT CHR$(scelta):PRINT "Hai battuto ";CHR$(scelta)
```

Si può ben comprendere come un programma con righe contenenti più istruzioni come questo non possa essere letto facilmente.

La sezione che segue fornisce le funzioni messe a disposizione dal BBC BASIC per semplificare lo sviluppo e la modifica di un programma. Fino ad ora si sono utilizzati programmi composti unicamente da una sequenza di istruzioni. Il computer può però anche ripetere le istruzioni, oppure scegliere tra diverse istruzioni quella da eseguire. Tutti i programmi sono composti da una combinazione di tre strutture di programma: *sequenza, ripetizione* e *scelta*. Le sezioni successive descriveranno come utilizzare queste strutture nel BBC BASIC.

Pianificazione della programmazione

I programmi esaminati nelle sezioni precedenti erano piuttosto brevi, ma il miglior modo per scrivere programmi più complessi è di strutturarli in modo diverso.

Esaminare il programma "faccia" visto in precedenza. Il programma consiste di una serie di istruzioni eseguite una dopo l'altra dal computer secondo l'ordine numerico delle righe. Un programma più lungo potrebbe contenere diverse centinaia di righe. Per scrivere un programma così lungo il modo più semplice è quello di spezzarlo in piccole parti chiamate **procedure**.

Il programma che segue mostra l'uso delle procedure:

```
1 REM disegno di una farfalla
10 MODE 130
20 PROCcorpo
30 PROCala_sinistra
40 PROCala_destra
50 END
60 DEFPROCcorpo
70 GCOL0,2
80 MOVE 640,500
90 MOVE 700,500
100 PLOT 205,640,700
110 ENDPROC
120 DEFPROCala_sinistra
130 GCOL 0,1
140 MOVE 200,200
150 DRAW 600,500
160 PLOT 85,200,800
170 ENDPROC
180 DEFPROCala_destra
190 GCOL 0,1
200 MOVE 1080,200
210 DRAW 680,300
220 PLOT 85,1080,800
230 ENDPROC
```

Il programma principale è costituito dalle righe da 10 a 50:

```
10 MODE 130
20 PROCcorpo
```

```
30 PROCala_sinistra
40 PROCala_destra
50 END
```

Le righe da 20 a 40 sono **richiami di procedura**. Ogni istruzione PROC ordina al computer di non eseguire la riga immediatamente successiva, ma di cercare all'interno del programma un'istruzione DEFinizione della PROCedura (DEFPROC) con il nome di procedura corretto e di eseguire l'istruzione in quella procedura.

Ad esempio, dopo la riga 20 il computer passa alla riga 60, quindi esegue le righe da 70 a 100 che disegnano il corpo della farfalla. La riga 110 termina la procedura (ENDPROC).

Dopo aver eseguito la procedura, il computer ritorna alla riga successiva al richiamo di procedura per continuare l'esecuzione del programma. In questo caso la riga subito dopo la riga 20 è un altro richiamo di procedura. *PROCala_sinistra* disegna l'ala sinistra della farfalla, mentre *PROCala_destra* disegna l'ala destra.

L'istruzione END nella riga 50 indica al computer che il programma è terminato. In alcuni programmi END è facoltativo, come ad esempio nel programma "faccia". In questo caso però è indispensabile perchè altrimenti il computer eseguirebbe la riga 60 e disegnerebbe nuovamente il corpo della farfalla.

L'ordine in cui appaiono le procedure non ha alcuna importanza. Le procedure possono essere inserite in qualsiasi punto del programma, tranne che all'inizio. Per l'assegnazione dei nomi alle procedure seguire le regole dei nomi di variabile, con l'unica eccezione che un nome di procedura può iniziare con un numero.

All'interno di un programma una stessa procedura può essere chiamata più volte, per evitare di ripetere righe di programma:

```
10 MODE 130
20 PROCvariabili
30 PROClocomotiva
40 END
50 DEFPROCvariabili
60 scala=0.6
70 xinizio=300
80 yinizio=200
90 xsportello=360
100 xdist=900
110 ysportello=500
120 ydist=300
130 xresping=300*scala
140 raggio_resping=50*scala
150 xfumaiolo=100*scala
160 yfumaiolo=200*scala
170 iniziofumaiolo=60*scala
180 xiniziosportello=xinizio+(xdist*scala)
190 xfinest=xiniziosportello-30*scala
200 yfinest=yinizio+(ydist-30)*scala
210 xdistfinest=300*scala
220 ydistfinest=200*scala
```

```
230 dist__ruota=130*scala
240 raggio__ruota=100*scala
250 ENDPROC
260 DEFPROClocomotiva
270 PROCrettangolo(xinizio,yinizio,xdist*scala,ydist*scala,1)
280 PROCrettangolo(xiniziosportello,yinizio,-xsportello*scala,
    ysportello*scala,1)
290 PROCrettangolo(xfinest,yfinest,-xdistfinest,ydistfinest,6)
300 PROCrettangolo(xinizio+iniziofumaiolo,yinizio+(ydist* scala),xfumaiolo,
    yfumaiolo,1)
310 PROCcerchio(xinizio+dist__ruota,yinizio,raggio__ruota,4)
320 PROCcerchio(xiniziosportello-dist__ruota,yinizio, raggio__ruota,4)
330 PROCcerchio(xinizio+xresping,yinizio+(ydist*scala), raggio__resping,1)
340 ENDPROC
350 DEFPROCrettangolo(x,y,xspostam,yspostam,col)
360 GCOL 0,col
370 MOVE x,y
380 PLOT 97,xspostam,yspostam
390 ENDPROC
400 DEFPROCcerchio(x,y,raggio,col)
410 GCOL 0,col
420 MOVE x,y
430 PLOT 153,raggio,0
440 ENDPROC
```

Le procedure per disegnare il motore utilizzano i comandi relativi MOVE e DRAW. *PROCvariabili* imposta i valori delle variabili utilizzate nel corso dell'intero programma. Le dimensioni e la posizione del motore potranno essere modificate cambiando i valori di *scala, xinizio* e *yinizio.*

I dati possono essere passati ad una procedura dal programma principale:

```
10 MODE 130
20 PROCcerchio(400,300,200)
30 PROCcerchio(600,600,100)
40 PROCcerchio(690,750,50)
50 END
60 DEFPROCcerchio(xcentro%,ycentro%,raggio%)
70 MOVE xcentro%,ycentro%
80 PLOT 157,xcentro%+raggio%,ycentro%
90 ENDPROC
```

I valori tra parentesi nella riga 20 sono chiamati **parametri.** Il computer memorizza i parametri nelle variabili *xcentro%, ycentro%* e *raggio%* nella riga 60, dove esegue il richiamo di procedura. Nella parte rimanente della procedura il computer utilizza queste variabili per disegnare un cerchio con il centro a 400,300 e con un raggio di 200.

Le righe 30 e 40 dimostrano come la stessa procedura possa essere utilizzata ogni volta che si disegna un cerchio, modificando semplicemente i parametri a seconda delle necessità.

Una procedura quale *PROCcerchio* è molto utile in quanto:

- può essere utilizzata molte volte all'interno dello stesso programma con diversi parametri per avere diversi risultati;
- può essere utilizzata anche se non si sa o non ci si ricorda come funziona la procedura;
- può essere utilizzata in altri programmi.

Per mantenere le coordinate del centro dello schermo in un programma si possono utilizzare *xcentro%* e *ycentro%*. Potrebbe sembrare che queste coordinate vengano perse se si utilizza *PROCcerchio* nello stesso programma in quanto anche questa istruzione possiede variabili chiamate *xcentro%* e *ycentro%*.

```
10 MODE 130
15 xcentro%=640:ycentro%=512
20 PROCcerchio(400,300,200)
30 PROCcerchio(600,600,100)
40 PROCcerchio(690,750,50)
45 PRINT"xcentro% rimane ";xcentro%
46 PRINT"ycentro% rimane ";ycentro%
50 END
60 DEFPROCcerchio(xcentro%,ycentro%,raggio%)
70 MOVE xcentro%,ycentro%
80 PLOT 157,xcentro%+raggio%,ycentro%
90 ENDPROC
```

Eseguire (RUN) il programma. I valori di *xcentro%* e *ycentro%* non sono influenzati da *PROCcerchio*. Questo perchè i parametri passati ad una procedura sono **locali** rispetto a quella procedura. *xcentro%*, *ycentro%* e *raggio%* in *PROCcerchio* esistono solo all'interno della procedura e non modificano i valori delle variabili che hanno lo stesso nome nel programma.

Tutte le variabili ad eccezione dei parametri sono **globali** per un programma. L'intero programma, comprese le procedure, conosce il valore delle variabili:

```
10 MODE 135
20 PROCnome
30 PROCstampa
40 END
50 DEFPROCnome
60 INPUT"Come ti chiami ",nome$
70 ENDPROC
80 DEFPROCstampa
90 PRINT"Questa procedura si chiama PROCstampa"
100 PRINT"Sa che ti chiami ";nome$
110 ENDPROC
```

La variabile stringa nome$ è globale, viene impostata con *PROCnome*, ma anche *PROCstampa* conosce *nome$* ed utilizza questo elemento.

La distinzione tra variabili locali e variabili globali diventa importante solo se una procedura contiene variabili globali. Di seguito viene mostrato un esempio di una procedura che centra il testo su una riga:

```
100 DEFPROC(testo$)
110 lunghezza%=LEN(testo$)
120 posizione_x%=(40-lunghezza%)/2
30 PRINT TAB(posizione_x%) testo$
40 ENDPROC
```

Questa è una procedura molto utile, che può essere richiamata diverse volte all'interno di uno stesso programma. La procedura comunque contiene due variabili globali, *lunghezza%* e *posizione_x%*. Se all'interno dello stesso programma vengono utilizzate variabili con lo stesso nome, il loro valore verrà perso al richiamo di *PROCcentro*:

```
10 MODE 135
20 lunghezza%=5
30 posizione_x%=15
40 PRINT"la lunghezza% è ";lunghezza%
50 PRINT"la posizione_x è ";posizione_x%
60 PROCcentro("Alcuni caratteri")
70 PRINT"la lunghezza% è ora ";lunghezza%
80 PRINT"la posizione_x% è ora ";posizione_x%
90 END
100 DEFPROCcentro(testo$)
110 lunghezza%=LEN(testo$)
120 posizione_x%=(40-lunghezza%)/2
130 PRINT TAB(posizione_x%) testo$
140 ENDPROC
```

Per fare in modo che le variabili all'interno di una procedura non interferiscano con la parte rimanente di programma, occorre dichiarare come *locali* le variabili. Aggiungere la riga riportata di seguito al programma precedente ed eseguirlo:

```
105 LOCAL lunghezza%,posizione_x%
```

In questo caso *lunghezza%* e *posizione_x%* rimangono invariati nonostante PROCcentro. Vi sono due copie di variabili: i valori globali, disponibili per l'intero programma, ed i valori locali, presenti esclusivamente all'interno di *PROCcentro*.

PROCcentro è ora completamente isolato e può essere utilizzato in un qualsiasi programma senza il problema di effetti inattesi.

Notare che le variabili possono essere utilizzate come parametri. Nel breve programma che segue *PROCcerchio* viene modificato in modo da permettere la selezione del colore da utilizzare:

```
10 MODE 135
20 PROCscelta
```

```
30 MODE 130
40 PROCcerchio(xscelta%,yscelta%,scelta_raggio,scelta_colore%)
50 END
60 DEFPROCscelta
70 INPUT"Centro del cerchio ",xscelta%,yscelta%
80 INPUT"Raggio ",scelta_raggio%
90 INPUT"Numero colore (da 1 a 15) ", scelta_colore%
100 ENDPROC
110 DEFPROCcerchio(xcentro%,ycentro%,raggio%,colore%)
120 GCOL 0,colore%
130 MOVE xcentro %,ycentro%
140 PLOT 157,xcentro%+raggio%,ycentro%
150 ENDPROC
```

Nel corso del presente capitolo relativo al BBC BASIC, le procedure vengono utilizzate per esteso, questo per la semplicità di scrittura e modifica di programmi suddivisi in piccole porzioni. Alcune delle procedure saranno specifiche per un particolare programma, mentre altre, quali ad esempio *PROCcerchio*, verranno utilizzate per più programmi, compresi i programmi scritti dall'utente.

Funzioni

Le funzioni sono routine che utilizzano uno o più parametri per calcolare un risultato. Il BBC BASIC contiene alcune funzioni incorporate. Battere quanto segue:

PRINT LEN"Computer Olivetti Prodest" **RETURN**

LEN è una funzione che utilizza una stringa come parametro e produce quale risultato la lunghezza della stringa. Digitare:

PRINT SQR 9 **RETURN**

SQR è una funzione che utilizza un numero quale parametro e come risultato dà la radice quadrata di questo numero.

Il BBC BASIC permette di impostare proprie funzioni, come mostrato nell'esempio seguente:

```
10 MODE 135
20 PROCinserimento_ora
30 END
40 DEFPROCinserimento_ora
50 PRINT'"Inserire ora espressa in minuti e secondi."
60 PRINT'"La funzione lo converte in"
70 PRINT"secondi."
80 INPUT'"Quanti minuti e secondi ",minuti%,secondi%
90 totale%=FNconversione(minuti%,secondi%)
100 PRINT'"Cioè ";totale%;" secondi."
110 ENDPROC
120 DEFFNconversione(min%,sec%)
130 =min%*60+sec%
```

La riga 90 richiama la funzione. Il computer ricerca DEFFN (definizione della funzione) all'interno dell'intero programma e lo trova alla riga 120.

La riga 130 inizia con un simbolo di uguale. Questo indica al computer che il calcolo successivo produrrà il risultato richiesto e che la funzione termina a questa riga. Il calcolo viene eseguito, la funzione termina ed il programma ritorna alla riga 90 e memorizza il risultato in *totale%*.

Questo esempio è stato utilizzato a solo scopo dimostrativo. Sarà molto più semplice, per raggiungere lo stesso scopo, utilizzare la riga di programma seguente:

```
90 totale%=minuti%*60+secondi%
```

Il programma che segue utilizza una funzione molto più complessa, che contiene istruzioni descritte nelle prossime sezioni:

```
10 MODE 135
20 PROCinserimento_parola
30 END
40 DEFPROCinserimento_parola
50 INPUT"Digitare una parola ",parola$
60 PRINT"Un anagramma di questa parola è ";FNanagramma(parola$)
70 ENDPROC
80 DEFFNanagramma(scelta$)
90 lunghezza%=LEN(scelta$)
100 FOR conteggio=1 TO lunghezza%
110 lettera_casuale%=RND(lunghezza%-1)
120 scelta$=RIGHT$(scelta$,lunghezza%-lettera_casuale%)+MID$
    (scelta$,lettera_casuale%,1)+LEFT$(scelta$,lettera_casuale%-1)
130 NEXT
140 =scelta$
```

Loop

FOR...NEXT

La caratteristica peculiare dei computer è data dalla loro capacità di ripetere le istruzioni. Grazie a questo è possibile trasformare programmi insignificanti in modo che possano produrre risultati sorprendenti.

Il loop FOR...NEXT permette di ripetere una serie di istruzioni per il numero di volte predeterminato:

```
10 MODE 128
20 FOR conteggio=1 TO 100
30 PRINT conteggio
40 NEXT conteggio
```

La riga 20 rappresenta l'inizio del loop, con il *conteggio* di variabile impostato inizialmente su 1. Dopo la visualizzazione del valore di *conteggio* con la riga 30, il computer ricerca l'istruzione NEXT che contrassegna la fine del loop.

A questo punto il conteggio viene aumentato di 1. Il computer ripete ora tutte le istruzioni, purchè il conteggio non abbia superato il valore 100 finale.

La riga 40 potrebbe essere scritta semplicemente come segue:

```
40 NEXT
```

L'uso del nome di variabile è opzionale, ma sarebbe consigliabile utilizzarlo in caso si inseriscano più loop all'interno di uno stesso programma, in modo da facilitarne la lettura.

Il valore di step può essere modificato in modo che gli incrementi non siano di una sola unità. Digitare:

```
20 FOR conteggio=7 TO 50 STEP 2
```

Il valore di step può contenere anche dei decimali:

```
20 FOR conteggio=3 TO 10 STEP 1.6
```

Questo valore può anche essere negativo, sebbene i valori di inizio e di fine di un loop devono allo scopo essere modificati in modo che il loop inizi con il valore più alto:

```
20 FOR conteggio=20 TO 1 STEP –1
```

Ovviamente i valori del loop possono essere anche variabili. Provare ad aggiungere al programma le righe riportate di seguito ed eseguire il programma alcune volte:

```
15 INPUT "Specificare valore iniziale",inizio
16 INPUT "Specificare valore finale",fine
17 INPUT "Specificare step",step
20 FOR conteggio=inizio TO fine STEP step
```

Per cancellare il programma memorizzato in seguito all'esempio precedente digitare **BREAK**.

Con l'esempio di programma seguente si può facilmente comprendere il potenziale di un loop:

```
10 MODE 130
20 PROCarte_moderna
30 END
40 DEFPROCarte_moderna
50 FOR conteggio=1 TO 50
60 PROCcerchio(RND(1279),RND(1023),RND(200),RND(7))
70 NEXT conteggio
80 ENDPROC
90 DEFPROCcerchio(xcentro%,ycentro%,raggio%,colore%)
100 GCOL 0,colore%
110 MOVE xcentro%,ycentro%
120 PLOT 157,xcentro%+raggio%,ycentro%
130 ENDPROC
```

RND produce un numero intero casuale da uno al valore racchiuso tra parentesi. La riga 60 traccia un cerchio con dimensioni casuali, in una posizione casuale e con un colore casuale richiamando *PROCcerchio* con parametri casuali.

Eseguire il programma riportato di seguito per meglio comprendere il funzionamento del programma Welcome:

```
50 FOR conteggio=7 TO 1 STEP -1
60 PROCcerchio(640,512,conteggio*50,conteggio)
```

Digitare:

```
10 MODE 128
RUN RETURN
```

per ritornare al modo 128.

E' possibile includere un loop FOR...NEXT in un altro loop FOR...NEXT. Questi loop vengono chiamati **loop nidificati**.

```
10 MODE 128
20 PROCtabelle
30 END
40 DEFPROCtabelle
50 FOR tabella=1 TO 12
60 PRINT""TAB(8)"La tabella di moltiplicazione""
70 FOR conteggio=1 TO 10
80 PRINT conteggio;" per ";tabella; " = ";conteggio*tabella
90 NEXT conteggio
100 PROCintroduzione
110 NEXT tabella
120 ENDPROC
130 DEFPROCintroduzione
140 PRINT """Premere un tasto quando si è pronti"
150 PRINT""TAB(2)"per la prossima tabella di moltiplicazione"""
160 tasto=GET
170 CLS
180 ENDPROC
```

Il loop principale, nelle righe da 50 a 110, esegue i conteggi delle tabelle di moltiplicazione da 1 a 12. L'altro loop, dalla riga 70 alla riga 90, è completamente nidificato all'interno del loop principale. Il loop nidificato moltiplica il valore della tabella per tutti i numeri da 1 a 10.

L'effetto dell'istruzione LIST può essere modificato per produrre automaticamente delle rientranze per tutte le coppie di istruzioni FOR e NEXT (e per alcune altre strutture). Battere:

```
LISTO 7 RETURN
LIST RETURN
```

Notare che l'inizio e la fine dei loop sono allineati verticalmente, questo per facilitare l'identificazione dei loop e la rilevazione di eventuali errori.

Cancellare la riga 90 che contiene un'istruzione NEXT e listare (LIST) nuovamente il programma. L'inizio (riga 50) e la fine (riga 110) del loop non sono più allineati verticalmente. Questo indica che in qualche punto del programma è presente un loop senza un'istruzione FOR o NEXT.

L'opzione fornita da LIST0 rimane attiva finché non si esegue un'operazione di ripristino (per mezzo di LIST0 0), un'interruzione hardware o si spegne il computer. Per il momento questa opzione verrà lasciata attiva per la sezione successiva.

REPEAT...UNTIL

Immaginiamo un programma basato sul gioco *Mastermind*. Il computer dovrà porre domande ripetutamente fino al raggiungimento del limite di un minuto. Si potrà utilizzare un loop FOR...NEXT in questo caso?

In questo caso non si potrà utilizzare un loop FOR...NEXT in quanto questo loop termina sempre quando viene raggiunto un certo valore quale risultato di un conteggio. Nel caso del Mastermind però non è possibile sapere in anticipo quante saranno le risposte date nell'arco di un minuto. Una persona entro questo termine potrebbe rispondere a tre domande, mentre un'altra potrebbe rispondere anche a dodici domande.

In questo caso si dovrà utilizzare un tipo di loop diverso, cioè il loop REPEAT...UNTIL. Questo loop termina quando una condizione viene soddisfatta, e non quando si raggiunge il risultato prefissato. Ad esempio, molti programmi prevedono una procedura che arresta l'esecuzione del programma fino alla pressione di un tasto:

```
10 MODE 129
20 PROCattesa
30 PRINT"Fine arresto esecuzione"
40 END
50 DEFPROCattesa
60 PRINT TAB(0,24)"Premere C per continuare"
70 REPEAT
80 tasto$=GET$
90 UNTIL tasto$="C"
100 ENDPROC
```

Le righe da 60 a 80 provocano l'attesa da parte del programma della pressione del tasto C. Premere un tasto diverso da C all'esecuzione della riga 70. Il computer rileva che *tasto$* non soddisfa la condizione della riga 80 e quindi esegue nuovamente il loop a partire dalla riga 60.

Il programma *Mastermind* potrebbe essere scritto come segue:

```
10 MODE 129
20 PROCgioco
30 END
40 DEFPROCgioco
50 TIME=0
60 risposta=0
70 REPEAT
80 primo=RND(12)
90 secondo=RND(12)
```

```
100 PRINT'"Quanto fa ";primo;" X ";secondo;
110 INPUT valorerisposta
120 risposta=risposta+1
130 UNTIL TIME>=6000
140 PRINT'"Hai risposto a ";risposta;" domande"
150 ENDPROC
```

La riga 50 introduce l'ora, cioè il valore dell'orologio interno del computer. TIME esegue conteggi in centesimi di secondo a partire da quando il computer viene acceso o ripristinato. La riga 50 riporta l'ora a zero, in modo che questo valore possa essere utilizzato per il conteggio dei minuti a disposizione per rispondere alle domande.

La risposta variabile viene utilizzata per calcolare il numero di risposte date e viene inizialmente impostata a zero dalla riga 60. Il loop viene eseguito dalla riga 70 alla riga 130, e pone ripetutamente domande di moltiplicazione casuale, finché TIME è >= (maggiore di o uguale a) 6000 centesimi di secondo, cioè un minuto.

Questo programma ha però una pecca. Al contrario di un conduttore di un programma televisivo a quiz non verifica le risposte. Per fare in modo che il programma esegua questa verifica, vedere la sezione seguente. Prima di continuare occorre salvare il programma per poterlo utilizzare successivamente.

Per salvare il programma digitare:

```
*DIR
SAVE "GIOVANNA"
```

In questo modo il programma viene salvato nel file chiamato "GIOVANNA".

Scelte

Come si è visto, il computer agisce in base ad una serie di istruzioni oppure ripete le istruzioni per il numero di volte specificato. Il computer può inoltre decidere se rispondere o meno ad un'istruzione:

```
10 MODE 129
20 PROCintroduzione
30 END
40 DEFPROCintroduzione
50 INPUT "Quanti anni hai ",eta'
60 IF eta'<=17 THEN PRINT'"Allora non puoi votare."
70 ENDPROC
```

Eseguire (RUN) il programma alcune volte, introducendo età diverse. Nella riga 60 il computer verifica l'istruzione dopo IF e, se è vera, esegue le istruzioni dopo THEN. Se l'istruzione è falsa, il computer ignora la parte rimanente di IF...THEN ed esegue la riga successiva.

Aggiungere le righe seguenti al programma ed eseguirlo diverse volte, fino a comprendere a fondo il funzionamento di IF...THEN:

```
63 IF età=32 THEN PRINT'"Hai la mia stessa età!"
66 IF età<65 THEN PRINT'"Non sei ancora in età pensionabile."
```

Il programma Mastermind può a questo punto essere ampliato in modo che possa verificare le risposte. Per richiamare il programma dalla memoria digitare:

```
LOAD "GIOVANNA"
LIST
```

Le righe nuove sono la 65, la 115 e la 145:

```
10 MODE 129
20 PROCgioco
30 END
40 DEFPROCgioco
50 TIME=0
60 risposta=0
65 sbagliato=0
70 REPEAT
80 primo=RND(12)
90 secondo=RND(12)
100 PRINT'"Quanto fa ";primo;" per ";secondo;
110 INPUT valorerisposta
115 IF valorerisposta<>primo*secondo THEN sbagliato=sbagliato+1
120 risposta=risposta+1
130 UNTIL TIME>=6000
140 PRINT'"Hai risposto a ";risposta;" domande"
145 PRINT'"Hai sbagliato ";sbagliato;" volte"
150 ENDPROC
```

La riga 115 può essere modificata in modo da fornire la risposta esatta e da calcolare le risposte sbagliate:

```
115 IF risposta<>primo*secondo THEN sbagliato=sbagliato+1:
PRINT "No, la risposta è ";primo*secondo
```

Può essere utilizzata una forma ampliata di IF...THEN in quanto i risultati possibili sono due: giusto o sbagliato:

```
115 IF risposta<>primo*secondo THEN sbagliato=sbagliato+1:
PRINT "No, la risposta è ";primo*secondo ELSE PRINT "Esatto!"
```

In altre parole, se (IF) la risposta è sbagliata, allora (THEN) il computer fornisce la risposta corretta, oppure (ELSE) si congratula per la risposta esatta.

La riga di programma è diventata piuttosto lunga. Per rendere più semplice la comprensione del programma, vediamo ora come ridimensionare la lunghezza della riga:

```
115 IF risposta<>primo*secondo THEN PROCsbagliato ELSE
PROCgiusto
```

e alla fine aggiungere:

```
160 DEFPROCsbagliato
170 sbagliato=sbagliato+1
```

```
180 PRINT "No, la risposta è ";primo*secondo
190 ENDPROC
200 DEFPROCgiusto
210 PRINT "Esatto"
220 IF sbagliato<2 THEN PRINT "Continua"
230 ENDPROC
```

Condizioni

Un loop REPEAT...UNTIL può essere impostato in modo che termini dopo il verificarsi di condizioni particolari:

```
10 MODE 129
20 PROCreazione
30 PROCtest
40 PROCcommento
50 END
60 DEFPROCreazione
70 PRINT TAB(0,8)"Premere il tasto corretto quando"
80 PRINT"lampeggia sullo schermo."
90 PRINT TAB(0,13)"Hai a disposizione 2 secondi e"
100 PRINT"puoi sbagliare fino a due volte oppure"
110 PRINT"dopo 20 secondi si spegne."
120 PRINT TAB(0,24)"Premi un tasto quando sei pronto.";
130 tasto=GET
140 ENDPROC
150 DEFPROCtest
160 CLS
170 sbagliato=0
180 giusto=0
190 TIME=0
200 REPEAT
210 lettera=RND(26)+64
220 PRINTTAB(19,11)CHR$(lettera)
229 REM VDU7 emette un suono
230 VDU7
240 risposta=INKEY(200)
250 IF risposta=-1 THEN sbagliato=sbagliato+1
260 IF risposta=lettera THEN giusto=giusto+1
270 UNTIL sbagliato=2 OR TIME>2000
280 ENDPROC
290 DEFPROCcommento
300 CLS
310 PRINT"Hai risposto correttamente ";giusto; "volte"
320 PRINT"Hai sbagliato ";sbagliato;" volte"
330 IF giusto>10 THEN PRINT"Bravissimo."
340 IF giusto<4 THEN PRINT"Il risultato è piuttosto scarso."
350 ENDPROC
```

Il loop esegue le righe da 200 a 270 e termina quando vengono date due risposte errate o dopo 20 secondi se non viene data alcuna risposta. In un loop IF...THEN può essere utilizzata l'istruzione OR:

```
325 IF giusto>10 OR sbagliato=0 THEN PRINT'"Bravissimo."
```

Per eseguire il loop IF...THEN solo se più condizioni si verificano contemporaneamente, digitare quanto segue:

```
345 IF giusto<3 AND sbagliato=2 THEN PRINT'"Patetico."
```

Per concludere un loop REPEAT...UNTIL può essere utilizzata l'istruzione AND

```
270 UNTIL sbagliato=2 AND giusto=5
```

In questo modo il loop termina dopo due risposte errate e cinque esatte.

Non vi è quasi limite al numero delle condizioni. Ad esempio si potrebbe avere:

```
270 UNTIL sbagliato=2 OR TIME>2000 OR giusto>5
```

Scelte multiple

IF...THEN...ELSE è utile quando vi sono solamente due scelte di azione. Spesso però in un programma vi possono essere più scelte. Ad esempio molte volte i programmi contengono un **menu** che permette all'utente di scegliere all'interno di una serie di azioni. Di seguito viene mostrato l'inizio di un programma di disegno contenente un menu:

```
10 MODE 129
20 PROCmenu
30 END
40 DEFPROCmenu
50 REPEAT
60 CLS
70 PRINT TAB(7,5)"Desideri:"
80 PRINT TAB(8,9)"1 caricare una figura"
90 PRINT TAB(8,12)"2 salvare una figura"
100 PRINT TAB(8,15)"3 disegnare una figura"
110 PRINT TAB(8,18)"4 terminare la procedura"
120 PRINT TAB(7,22)"Scegli un numero da 1 a 4";
130 REPEAT
140 risposta=GET
150 UNTIL risposta>42 AND risposta<53
160 scelta=risposta-48
170 ON scelta PROCcaricamento, PROCsalvataggio, PROCdisegno,
    PROCverifica
180 UNTIL scelta=4
190 ENDPROC
```

(Questo programma è incompleto e, se eseguito, dà un messaggio di errore).

Il loop dalla riga 130 alla riga 150 termina solo quando si preme un tasto con un codice ASCII da 48 a 53. Questi sono i codici ASCII dei numeri da 1 a 4 sulla tastiera, quindi questo loop non visualizza tasti premuti erroneamente, quali ad esempio Q o W.

Sottraendo il valore 48 dal codice ASCII nella riga 160 si ottiene nuovamente un numero da 1 a 4 e la riga 170 utilizzerà questo numero per scegliere quale delle quattro procedure eseguire. Se *scelta*=1, il computer eseguirà *PROCcaricamento*, se *scelta*=2 eseguirà *PROCsalvataggio* ecc. Al termine della procedura l'esecuzione continuerà a partire dalla riga **180**.

Il programma evita il problema che potrebbe insorgere dalla pressione di un tasto errato proseguendo l'esecuzione solo dopo la pressione dei tasti da 1 a 4. In ogni caso è possibile omettere il loop dalla riga 130 alla riga 150 in modo che il problema dei tasti errati venga gestito grazie ad una modifica all'interno dell'istruzione ON...PROC, come mostrato di seguito:

```
120 PRINT TAB(7,22)"Scegli un numero da 1 a 4 ";
140 risposta=GET
160 scelta=risposta-48
170 ON scelta PROCcaricamento, PROCsalvataggio, PROCdisegno,
    PROCverifica ELSE PROCtasto_sbagliato
```

Il computer esegue *PROCtasto_sbagliato* se la scelta non cade nella gamma da 1 a 4. L'istruzione ELSE può essere seguita da un'istruzione singola, non necessariamente PROC. Ad esempio:

```
170 ON scelta PROCcaricamento, PROCsalvataggio, PROCdisegno,
PROCverifica ELSE PRINT"Tasto sbagliato!"
```

ON...PROC è molto utile, ma funziona esclusivamente con numeri superiori a 1, a passi di 1. Generalmente si rende necessario eseguire un certo tipo di calcolo per poter produrre una gamma corretta di valori.

Gestione errori

Per ridurre il tempo impiegato per la correzione di errori nei programmi vengono utilizzate procedure e nomi di variabili sensate, ma è inevitale commettere errori. Il computer è in grado di rilevare da solo alcuni tipi di errore, visualizzando un messaggio.

Si consiglia di includere sempre nei programmi una routine di gestione errori che informi l'utente dell'errore e ne faciliti la correzione.

```
10 ON ERROR GOTO50
20 MODE 130
30 PROCprogramma_princ
40 END
50 MODE 7
60 PRINT"Errore numero ";ERR;" sulla riga ";ERL
70 END
```

Questo programma contiene un grave errore: non esiste una procedura chiamata *PROCprogramma_princ*. L'esecuzione del programma darà il seguente risultato:

Error number 29 at line 30
(Errore numero 29 sulla riga 30)

L'istruzione ON ERROR sulla riga 10 indica al computer di tornare alla riga 50 se rileva un errore in fase di esecuzione del programma. Tutti gli errori rilevabili dal computer sono identificati da un numero memorizzato nella variabile ERR. ERL invece memorizza il numero della riga che contiene l'errore.

Il manuale di riferimento fornisce una lista completa dei numeri di errore e descrive in dettaglio ciascun errore. Comunque, è possibile richiedere al computer maggiori informazioni sull'errore verificatosi, grazie all'istruzione REPORT nella routine di gestione errori:

```
55 REPORT
60 PRINT " sulla riga ";ERL
```

L'esecuzione del programma darà il seguente risultato:

No such FN/PROC at line 30
(FN/PROC inesistente sulla riga 30)

Questo indica che il computer non ha trovato una procedura chiamata *PROCprogramma_princ* nella riga 30.

Ci si potrebbe chiedere, visto che il computer fornisce automaticamente i messaggi di errore, perché includere nel programma una routine di gestione errori. La ragione principale è data dal fatto che la routine può riportare il computer allo stato normale e, se non utilizzata, i messaggi potrebbero risultare illeggibili, come accade eseguendo il programma seguente:

```
10 MODE 2
20 VDU 28,19,31,19,0
30 COLOUR 135
40 un terribile errore
50 END
```

Aggiungere le righe seguenti per comprendere i vantaggi di una routine di rilevazione errori:

```
35 ON ERROR GOTO 60
60 MODE 7
70 REPORT
80 PRINT "sulla riga ";ERL
90 END
```

Ulteriori informazioni sulle stringhe

Le stringhe sono praticamente gruppi di caratteri. La sezione che segue tratta la gestione delle stringhe in BBC BASIC.

E' possibile unire (concatenare) diverse stringhe semplicemente odinandoal computer di aggiungere una stringa alla fine della stringa precedente, come segue:

```
10 MODE 129
20 primo$="L'inizio"
30 secondo$="e la fine."
40 tutto$=primo$+secondo$
50 PRINT tutto$
```

L'esecuzione del programma darà:

```
L'inizioe la fine.
```

Tranne per il fatto che una stringa composita non può superare i 255 caratteri, non esiste alcun limite al numero delle stringhe che possono essere concatenate. Ad esempio, per includere uno spazio addizionale nella riga di programma precedente, digitare:

```
40 tutto$=primo$+" "+secondo$
```

E' inoltre possibile confrontare due stringhe utilizzando <, = e > (o una combinazione dei tre simboli). Le due stringhe vengono confrontate carattere per carattere fino a quando viene rilevata una differenza. La stringa che contiene il carattere precedente in ordine alfabetico è minore dell'altra stringa. Ad esempio:

CIGNO è minore di SQUALO perchè C viene prima di S;

CIGNO è minore di TOPO perchè C viene prima di T;

CIGNO è maggiore di CANE perchè entrambe le parole iniziano per C, ma I viene dopo A;

CIGNO è maggiore di BALENA perchè C viene dopo B.

Per maggiore chiarezza, eseguire il programma seguente, che permette di confrontare coppie di stringhe:

```
10 MODE 129
20 REPEAT
30 INPUT LINE "Quale è la prima stringa", prima$
40 INPUT LINE "Quale è la seconda stringa", seconda$
50 IF prima$<seconda$ THEN PRINT prima$;" è precedente alfabeticamente
a ";seconda$
60 IF prima$=seconda$ THEN PRINT "Le due stringhe sono identiche."
70 IF prima$>seconda$ THEN PRINT prima$;" è successiva alfabeticamente
a ";seconda$
80 UNTIL prima$="STOP"
```

Notare l'uso di STOP per arrestare il programma.
Il computer può ordinare alfabeticamente una lunga lista di stringhe per mezzo del confronto mostrato qui sopra. In altre parole, il computer confronta i codici ASCII dei caratteri interessati. La lettera "a" minuscola è considerata maggiore della lettera "A" maiuscola in quanto il codice ASCII di "a" è 97, mentre per "A" è 65.

LEN permette di calcolare il numero di caratteri che compongono una stringa:

```
10 MODE 129
```

```
20 INPUT "Quale è la stringa ";scelta$
30 lunghezza=LEN(scelta$)
40 PRINT scelta$;" contiene ";lunghezza;" caratteri."
```

Il programma di anagramma visto precedentemente utilizzava LEN per trovare la lunghezza della parola introdotta, quindi risistemava i caratteri combinando parti della stringa in ordine diverso. Per copiare parti di stringa sono disponibili diverse funzioni: (Per cancellare il programma memorizzato in seguito all'esempio precedente digitare BREAK).

```
10 MODE 129
20 esempio$="Yellow submarine"
30 PRINT "La parola è ";esempio$
40 parte1$=LEFT$(esempio$,4)
50 PRINT "Le prime quattro lettere a sinistra sono ";parte1$
60 parte2$=RIGHT$(esempio$,6)
70 PRINT "Le ultime 6 lettere a destra sono ";parte2$
80 parte3$=MID$(esempio$5,6)
90 PRINT "Le 6 lettere centrali sono ";parte3$
100 parte4$=MID$(esempio$,4)
110 PRINT "Le lettere a partire dal quarto carattere sono: ";parte4$
```

LEFT$ e RIGHT$ funzionano in modo simile, prelevando il numero di caratteri specificato rispettivamente a sinistra o a destra della stringa. MID$ è leggermente diverso dai primi due e nella riga 80 viene utilizzato per prelevare le lettere a partire dalla quinta per un totale di 6 lettere. Nella riga 100 il secondo numero viene tralasciato, causando in questo modo il prelevamento da parte di MID$ di tutti i caratteri a partire dal quarto fino alla fine della stringa. Inutile dire che i numeri nell'esempio possono essere sostituiti da variabili numeriche.

Può essere creata una stringa che consiste in una serie di copie di un'altra stringa. Questa operazione viene effettuata tramite l'istruzione STRING$:

```
10 MODE 129
20 INPUT "Quale è la stringa ",testo$
30 copia$=STRING$(10,testo$)
40 PRINT "Una stringa contenente 10 copie ha questo aspetto:"
50 PRINT copia$
```

INSTR viene utilizzato per rilevare la prima volta in cui una stringa appare all'interno di un'altra stringa. Ad esempio:

```
10 MODE 129
20 INPUT LINE "Introdurre una frase",frase$
30 controllo=INSTR(frase$,"e")
40 PRINT "La frase";
50 IF controllo>0 THEN PRINT "contiene una e alla posizione ";controllo ELSE
PRINT "non contiene una 'e'"
```

La variabile *controllo* nella riga 30 contiene la posizione all'interno di *frase$* in cui appare per la prima volta la lettera "e". Se *frase$* non contiene una "*e*", *controllo* sarà 0. Con INSTR è possibile ricercare anche gruppi di lettere. Ad esempio, sostituendo la riga 30 con:

```
30 controllo=INSTR(frase$,"il")
```

provoca la ricerca di "*il*" in *frase$*.

Su una variabile stringa non possono essere eseguiti calcoli aritmetici, anche se la variabile stringa contiene solo caratteri numerici. In questi casi, per modificare un numero in stringa e viceversa, sono disponibili due funzioni:

```
10 MODE 129
20 INPUT "Che giorno è oggi ";numero
30 numero$=STR$(numero)
40 INPUT "In che mese siamo ";mese$
50 data$=mese$+" "+numero$
60 PRINT "Oggi è il ";data$
```

STR$ nella riga 30 converte *numero* nella variabile numerica in *numero$* di una variabile stringa. Le righe 50 e 60 vengono introdotte per mostrare come la versione stringa possa essere concatenata con altre stringhe.

VAL dà il valore numerico di una stringa:

```
10 MODE 129
20 INPUT "Battere una combinazione di numeri e lettere ";combinazione$
30 numero=VAL(combinazione$)
40 IF numero>0 THEN PRINT "La stringa inizia con i numeri "; numero
```

Se una stringa inizia con caratteri numerici, con un simbolo + o – , VAL converte questi caratteri nel loro equivalente numerico. Notare che VAL ignora la parte di stringa che segue il primo carattere non numerico rilevato. Ad esempio:

```
PRINT VAL("123g456") RETURN
```

darà come risultato 123.

READ, DATA e RESTORE

Molti programmi, prima di poter essere eseguiti, necessitano l'introduzione di alcuni dati fondamentali. Sarebbe consigliabile memorizzare questi dati come parte di programma. Come esempio, di seguito viene fornito un programma gioco che contiene domande e risposte nelle istruzioni DATA:

```
10 MODE 129
20 PROCinizio
30 PROCgioco
40 END
50 DEFPROCinizio
60 corretto=0
70 READ quante
80 PRINT TAB(14)"gioco"
90 PRINTSTRING$(40,"=")
100 ENDPROC
110 DEFPROCgioco
120 FOR domanda=1 TO quante
130 READ domanda$,risposta_giusta$
```

```
140 PRINT'domanda$
150 INPUT risposta$
160 IF risposta$=risposta_giusta$ THEN PROCgiusto
    ELSE PROCsbagliato
170 NEXT domanda
180 PRINT'"Hai risposto correttamente a ";corretto;" risposte su ";quante
190 ENDPROC
200 DEFPROCsbagliato
210 corretto=corretto+1
220 IF RND(2)>1 THEN PRINT'"Esatto!" ELSE PRINT'"Bravo!"
230 ENDPROC
240 DEFPROCsbagliato
250 PRINT'"No, la risposta è:"
260 PRINTrisposta_giusta$
270 ENDPROC
280 DATA3
290 DATAIn quale secolo siamo,20
300 DATAQuale re britannico ha avuto sei mogli,Enrico VIII
310 DATAQual'è il seme della quercia, ghianda
```

L'istruzione READ nella riga 70 causa la ricerca da parte del computer all'interno del programma, fino al ritrovamento della prima riga che inizi con la parola DATA, in questo caso la 280. Il computer legge il primo valore dopo la parola DATA e lo memorizza nella variabile *quante*. Gli elementi DATA possono essere numeri o stringhe e sono separati da virgole.

Il loop dalla riga 120 alla riga 170 viene eseguito 3 volte (valore di *quante*). La riga 130 legge quindi una domanda e risposta dalle istruzioni DATA. Ogni volta il computer legge i dati a partire dal punto in cui si era interrotto nel loop, quindi ogni volta leggerà una domanda e risposta diversa.

In un'istruzione DATA può essere introdotto un qualsiasi numero di elementi di dati fino a raggiungere il massimo di 255 caratteri per riga. Quindi, tutti i dati di un programma potrebbero essere contenuti in una sola riga:

```
280 DATA3,In quale secolo siamo,20,Quale re britannico ha avuto sei mogli,Enrico VIII,Qual'è il seme della quercia, ghianda
```

Il motivo principale per cui si sono suddivisi i dati è per potere eseguire più facilmente eventuali modifiche. Per motivi analoghi le righe DATA vengono generalmente raggruppate sebbene possano essere introdotte in qualsiasi punto del programma. Per aggiungere al programma una domanda supplementare, digitare le righe seguenti:

```
280 DATA4
320 DATAChi ha vinto il campionato mondiale di calcio nel 1984,Italia
```

L'esecuzione del programma solleva uno dei problemi che si possono incontrare utilizzando delle stringhe. Il computer accetta come corretta una risposta che corrisponda esattamente alla risposta memorizzata. Ad esempio, darà come sbagliata la risposta *Enrico ottavo* alla seconda domanda, in quanto nel programma era stato specificato Enrico VIII come risposta corretta.

E' possibile utilizzare l'istruzione RESTORE per leggere l'istruzione DATA a partire dalla riga specificata. Aggiungere le righe seguenti al programma gioco per comprendere risposte alternative alle domande:

```
91 PRINT "Desideri (1) una domanda generale"
92 PRINT TAB(13)"(2) o una domanda sugli animali"
93 INPUT "1 o 2",scelta
94 IF scelta=1 THEN RESTORE 280 ELSE RESTORE 500
500 DATA3
510 DATACome si chiama il piccolo di una gallina,pulcino
520 DATAQuale è il mammifero più grande,balena
```

La riga 94 utilizza RESTORE per leggere i dati a partire dalla riga 280 o 500, a seconda del tipo di domande scelto.

Matrici

Il computer è molto utile per ricercare un particolare elemento di dati in una lunga lista oppure per ordinare in modo particolare una serie di dati. Ad esempio, con il computer è possibile ordinare alfabeticamente una lista di nomi. Per eseguire questa operazione il computer deve confrontare uno con l'altro tutti i nomi per decidere quale viene prima in ordine alfabetico. Tutti i nomi devono essere accessibili contemporaneamente, quindi saranno più facili da confrontare se memorizzati tutti in una lista o **matrice**.

Il programma seguente legge 10 nomi in una matrice, quindi visualizza il nome selezionato:

```
10 MODE 129
20 PROCimpostazione_matrice
30 PROCricerca
40 END
50 DEFPROCimpostazione_matrice
60 DIM nome$(10)
70 FOR conteggio=1 TO 10
80 READ nome$(conteggio)
90 NEXT conteggio
100 ENDPROC
110 DATA Salvi,Bianchi,Rossi,Besana
120 DATA Tizi,Donghi,Verdi,Alberti,Longo,Bruni
130 DEFPROCricerca
140 INPUT "quale nome desideri (1–10)",numero
150 PRINT'"Il numero ";numero;" della lista è ";nome$(numero)
160 ENDPROC
```

L'istruzione DIM alla riga 60 comunica al computer il numero di elementi contenuti nella matrice, in questo caso 10. Il loop dalla riga 70 alla riga 90 legge i nomi dalle istruzioni di dati e li memorizza automaticamente nella matrice *nome$*, in modo che *nome$(1)* sia Salvi, *nome$(2)* sia Bianchi, ecc. PROCricerca nella riga 130 è stato incluso nel programma per avere una conferma che il computer abbia memorizzato tutti i nomi nell'ordine in cui erano stati introdotti nelle istruzioni DATA.

La ricerca all'interno di una matrice di un nome o una serie di nomi che rispondano a determinati requisiti viene eseguita dal programma molto velocemente. Per ricercare ad esempio tutti i nomi che inizino con una lettera particolare, modificare le ultime righe di programma come segue:

```
140 INPUT "Con quale lettera deve iniziare il nome ",lettera$
150 FOR conteggio=1 TO 10
160 nome$=nome$(conteggio)
170 IF LEFT$(nome$,1)=lettera$ THEN PRINT nome$
180 NEXT conteggio
190 ENDPROC
```

Questo programma contiene pochi nomi, ma il computer riesce a gestire liste composte anche da centinaia di nomi. L'unico limite è costituito dalla capacità di memoria del computer.

Nella maggior parte dei casi si lavora su più matrici contemporaneamente. Generalmente vengono create liste di elementi di dati che sono composte da nomi ed indirizzi, libri e autori, ecc. Ad esempio, memorizzando nomi ed età, occorrerà impostare due matrici. La relazione tra le diverse matrici permette al computer di eseguire molto più facilmente l'operazione di ricerca. Se Bianchi fosse il quinto nome della matrice, anche la sua *età* sarà nella quinta posizione della matrice:

```
nome$(5)= Bianchi" età(5)=27
```

In questo caso l'età è memorizzata in una matrice numerica età( ) invece che in una matrice di stringa in quanto l'operazione da svolgere sarà sull'età.

Il programma che segue memorizza i nomi e le età di 10 persone e ricerca all'interno della matrice l'età di ciascuna persona non appena si introduce il cognome:

```
10 MODE 129
20 PROCimpostazione_matrice
30 PROCricerca_anni
40 END
50 DEFPROCimpostazione_matrice
60 DIM nome$(10), anni(10)
70 FOR conteggio=1 TO 10
80 READ nome$(conteggio), anni(conteggio)
90 NEXT conteggio
100 ENDPROC
110 DATA Salvi,42,Bianchi,35,Rossi,57
120 DATA Cerri,35,Besana,49,Tizi,23
130 DATA Donghi,39,Verdi,24,Alberti,63,Longo,75
140 DEFPROCricerca_anni
150 INPUT "Di chi vuoi l'età ",ricerca$
160 conteggio=1
170 REPEAT
180 nome$=nome$(conteggio)
190 IF nome$=ricerca$ THEN PRINT nome$;" ha ";anni(conteggio);" anni"
200 conteggio=conteggio+1
```

```
210 UNTIL conteggio=11 OR nome$=ricerca$
220 IF nome$<>ricerca$ THEN PRINT ricerca$;" non è nella lista"
230 ENDPROC
```

Per impostare la dimensione di entrambe le matrici viene utilizzata soltanto un'istruzione DIM (riga 60). Il loop dalla riga 170 alla riga 210 esamina ogni nome nella matrice per trovare quello richiesto.

Nel programma possono anche essere utilizzate matrici intere. Nel programma precedente tutte le età erano numeri interi ed avrebbero potuto essere memorizzate in una matrice *età%( )*.

File

L'ultima sezione esaminata mostrava come memorizzare dati nelle matrici. Questo metodo di memorizzazione richiede grande spazio di memoria. Ogni elemento viene infatti memorizzato due volte: una volta come parte dell'istruzione DATA all'interno del programma ed un'altra volta in un altro punto della memoria quando il computer esegue la copia di tutti gli elementi nella matrice.

Un metodo più funzionale è dato dalla memorizzazione dei dati completamente separata dal programma, sotto forma **di file dati**. Il file può essere salvato su disco (esattamente come avviene per un programma) e può essere richiamato in qualsiasi momento.

Il programma che segue crea un file composto da nomi e numeri telefonici:

```
10 MODE 129
20 PROCprendere_nomi
30 PROCcreazione_file
40 END
50 DEFPROCprendere_nomi
60 DIM nome$(100), telefono$(100)
70 PRINT'"Introduci i nomi e"
80 PRINT"i numeri telefonici dei tuoi amici."
90 PRINT"Puoi terminare con XXX quando"
100 PRINT"ti viene richiesto un nome."
110 conteggio=0
120 REPEAT
130 conteggio=conteggio+1
140 INPUT "Nome",nome$(conteggio)
150 IF nome$(conteggio)<>"XXX" THEN INPUT "numero telefonico",
    telefono$(conteggio)
160 UNTIL nome$(conteggio)="XXX" OR conteggio=100
170 ENDPROC
180 DEFPROCcreazione_file
190 CLS
200 PRINT'"Quale nome desideri assegnare al"
210 INPUT"file dati",file$
220 questo=OPENOUT(file$)
230 IF nome$(conteggio)="XXX" THEN conteggio=conteggio-1
```

```
240 FOR numero=1 TO conteggio
250 PRINT#questo,nome$(numero),telefono$(numero)
260 NEXT numero
270 CLOSE#questo
280 ENDPROC
```

PROCprendere_nomi imposta due matrici stringa che possono contenere un massimo di 100 nomi e numeri telefonici. Il loop dalla riga 120 alla riga 160 riceve i dati introdotti da tastiera e memorizza i nomi ed i numeri in due matrici.

PROCcreazione_file crea il file, a cui viene assegnato il nome nella riga 210. La riga 200 apre il file per mezzo dell'istruzione OPENOUT in modo che questo possa ricevere i dati.

Il BBC BASIC permette di avere contemporaneamente cinque file aperti. Il computer assegna un numero a ciascun file, in modo da distinguere un file da un altro. Questo numero viene chiamato **numero di canale**. Tutti i riferimenti al file vengono fatti per mezzo del numero di canale. Per questo motivo è indispensabile che questo numero venga salvato. La riga 220 memorizza il numero di canale per il file nella variabile *questo*.

Il loop dalla riga 230 alla riga 250 scrive i dati nel file. La riga 240 ordina al computer di visualizzare i dati tramite il canale *questo*.

La riga 260 è stata introdotta per comunicare al computer che non vi sono più dati e quindi per chiudere il canale al termine della scrittura dell'intero file.

Notare che eseguendo il programma si provoca solo il salvataggio del file che contiene i nomi ed i numeri telefonici. Il programma stesso dovrà essere salvato nello stesso modo in cui viene salvato qualsiasi altro programma. Un file è inutile se non è possibile accedervi per leggere le informazioni in esso contenute. Il programma che segue memorizza i nomi ed i numeri telefonici contenuti nel file e ricerca il numero telefonico dell'amico specificato:

```
10 MODE 129
20 PROClettura_file
30 PROCricerca_numero
40 END
50 DEFPROClettura_file
60 DIM amico$(100), nome$(100)
70 PRINT"Quale nome hai assegnato al"
80 INPUT"file dati",file$
90 quello=OPENIN(file$)
100 conteggio=0
110 REPEAT
120 conteggio=conteggio+1
130 INPUT£quello,amico$(conteggio),numero$(conteggio)
140 UNTIL EOF£quello
150 CLOSE£quello
160 ENDPROC
170 DEFPROCricerca_numero
180 CLS
190 INPUT "Di chi desideri il numero",nome$
```

```
200 ricerca=0
210 REPEAT
220 ricerca=ricerca+1
230 IF nome$=amico$(ricerca) THEN PRINTnome$;" ha il numero ";numero$
    (ricerca)
240 UNTIL ricerca=conteggio OR nome$=amico$(ricerca)
250 IF nome$<>amico$(ricerca) THEN PRINT"Non trovo questo nome"
260 ENDPROC
```

PROClettura_file legge il contenuto del file e memorizza i nomi ed i numeri telefonici nelle due matrici *amico$( )* e numero$( ).

La riga 90 apre il file con l'istruzione OPENIN per poter introdurre i dati nel file stesso. Anche in questo caso occorre salvare il numero di canale, questa volta memorizzandolo nella variabile *quello*.

Il loop dalla riga 110 alla riga 140 legge i dati dal file e memorizza i dati nelle matrici. La riga 130 introduce i dati tramite il canale *quello*.

Il computer non conosce a priori il numero degli elementi contenuti nel file, quindi continua la lettura delle informazioni fino a raggiungere EOF (fine del file) nella riga 140. Al termine, la riga 150 chiude il file, in quanto non vi sono più dati da leggere.

Si sono così copiati tutti i dati dal file alle matrici *amico$( )* e *numero$( )*. Introducendo il nome di un amico, *PROCricerca_numero* ricercherà nelle matrici il numero di telefono di questa persona.

Gli ultimi due programmi sono molto semplici ed illustrano i principi di utilizzo dei file. Possono comunque essere scritti programmi molto più sofisticati per creare e modificare file di dati di qualsiasi natura. Il sistema può essere ampliato per poter utilizzare il programma già pronto VIEWSTORE, un programma Olivetti Prodest molto potente di gestione file.

Ulteriori informazioni sulle funzioni grafiche

Con il modo grafico è possibile visualizzare simultaneamente sullo schermo un numero fisso di colori puri. Possono inoltre essere visualizzati quattro altri colori composti da una combinazione di colori puri. Ad esempio, nel modo 129 sono disponibili quattro colori puri e quattro misti. Il programma seguente visualizza tutti gli otto colori contemporaneamente disegnando sette rettangoli su sfondo nero.

```
10 MODE 129
20 PROCpuro
30 PROCmisto
40 END
50 DEFPROCpuro
60 FOR colore=1 TO 3
70 GCOL0,colore
80 angolo=80*colore
90 PROCrettangolo(angolo,angolo,angolo+100,angolo+100)
100 NEXT colore
```

```
110 ENDPROC
120 DEFPROCmisto
130 FOR colore=16 TO 64 STEP 16
140 GCOL colore,0
150 angolo=80*((colore/16)+3)
160 PROCrettangolo(angolo,angolo,angolo+100,angolo+100)
170 NEXT colore
180 ENDPROC
190 DEFPROCrettangolo(x,y,x1,y1)
200 MOVE x,y
210 PLOT 101,x1,y1
220 ENDPROC
```

PROCpuro traccia rettangoli nei colori puri in seguito al comando GCOL 0 nella riga 70.

PROCmisto traccia rettangoli utilizzando i colori misti. Nel loop il colore misto viene scelto in base a quanto specificato nel comando GCOL nella riga 140. La prima volta sarà GCOL 16,0, la seconda GCOL 32,0 ecc.

I modelli impostati da GCOL 16,0 e dagli altri comandi GCOL di numero più alto non sono fissi e possono essere modificati da un comando VDU. Aggiungere le righe seguenti al programma ed eseguirlo:

```
121 REM sfumature gialle e nere per GCOL 16,0
122 VDU23,2,160,80,160,80,160,80,160,80
```

Il comando VDU 23,2 modifica il colore misto prodotto da GCOL 16,0. Gli otto numeri dopo VDU 23,2 definiscono il nuovo colore misto, in questo caso aree alternate gialle e nere. Allo stesso modo è possibile utilizzare VDU 23,3 per impostare un nuovo colore misto prodotto da GCOL 32,0 e VDU 32,4 per modificare il colore misto prodotto da GCOL 48,0. Aggiungere al programma le righe seguenti per ottenere un set completamente nuovo di colori misti:

```
123 REM grandi aree di sfumature rosse e gialle per GCOL 32,0
124 VDU23,3,60,195,60,195,60,195,60,195
125 REM sfumature nere e rosse per GCOL 48,0
126 VDU23,4,5,10,5,10,5,10,5,10
127 REM sfumature nere e bianche per GCOL 64,0
128 VDU23,5,85,170,85,170,85,170,85,170
```

La determinazione di quale degli otto numeri produce un particolare colore misto è piuttosto complessa e la procedura varia da modo a modo. Allo scopo viene utilizzato il generatore di colori misti (PFILL) del programma Welcome, descritto alla fine di questo capitolo. Per ulteriori dettagli sul funzionamento del comando VDU23 consultare il manuale di riferimento.

PFILL permette di definire propri colori e visualizza i numeri per ricreare questi colori. I numeri vengono visualizzati nel formato **esadecimale**. Non occorre conoscere alla perfezione la numerazione esadecimale. E' sufficiente inserire i numeri in un'istruzione VDU per poter utilizzare il colore misto all'interno del proprio programma. Ad esempio:

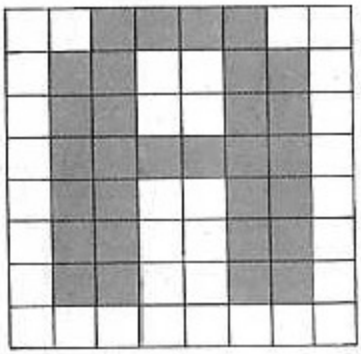
```
122 VDU23,2,&A0,&50,&A0,&50,&A0,&50,&A0,&50
```

è l'equivalente in esadecimale della riga 122 precedente ed ha lo stesso effetto (il simbolo & viene utilizzato per indicare che il numero che lo segue è in notazione esadecimale).

Definizione di caratteri personalizzati

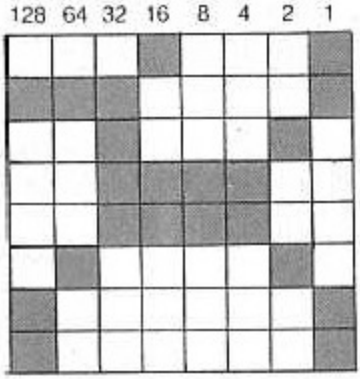
VDU 23 può essere utilizzato per creare nuovi caratteri per giochi o programmi scientifici o matematici che richiedano una notazione insolita. Come si ricorderà, il simbolo di *p greco* era stato utilizzato nell'esempio nell'introduzione di questo manuale.

Tutti i normali caratteri sono prodotti su una griglia di 8x8 punti (o pixel). La lettera A maiuscola ad esempio è rappresentata come segue:



Tutti i caratteri possono essere ridefiniti, ma non sarebbe consigliabile modificare una qualsiasi lettera dell'alfabeto per rappresentare un carattere particolare. Allo scopo sono disponibili i codici ASCII da 128 a 255 che non sono predefiniti, in modo da non modificare i caratteri alfabetici.

Di seguito viene mostrato un carattere con la forma di un cane tracciato sulla griglia 8x8:



Per ridefinire il carattere 255 in modo che diventi della forma mostrata qui sopra, battere:

VDU23,255,17,255,34,60,60,66,129,129

Per visualizzare il carattere, digitare:

MODE 129 RETURN
PRINT CHR$(255) RETURN

Ciascun numero dopo VDU 23,255 descrive una delle otto file di punti che, insieme, formano il carattere, partendo dall'alto. Per ottenere questo numero occorre prendere nota dei punti che compongono la fila che saranno illuminati quando viene visualizzata la figura. Ad esempio, nella fila superiore saranno illuminati solo il quarto e l'ultimo punto. Il numero per descrivere questa riga sarà quindi 16+1=17, ottenuto sommando le cifre al di sopra di questi due punti.

Allo stesso modo, la seconda fila sarà identificata come segue:

128+64+32+1=225, la terza riga 32+2=34, ecc.

Tracciare una griglia di 8x8 punti e provare a comporre un carattere.

Il programma Welcome contiene una routine di definizione carattere chiamata Definizione Carattere (CHARDES) che fornisce un metodo automatico per modificare i caratteri visualizzabili dal computer. Questa routine sarà descritta alla fine di questo capitolo.

Modifica della gamma di colori

I modi come il modo 128 possono visualizzare solo due colori contemporaneamente. Ad esempio, i colori normali per il modo 128 sono nero e bianco. Sebbene non si possano avere più di due colori contemporanemante, è possibile modificare la gamma di colori visualizzabili. In questo modo, al posto di nero e bianco si potrà avere rosso e giallo.

I numeri utilizzati nei comandi GCOL e COLOUR producono diversi effetti a seconda della modalità utilizzata ed anche il colore visualizzato dipende da due serie di dati.

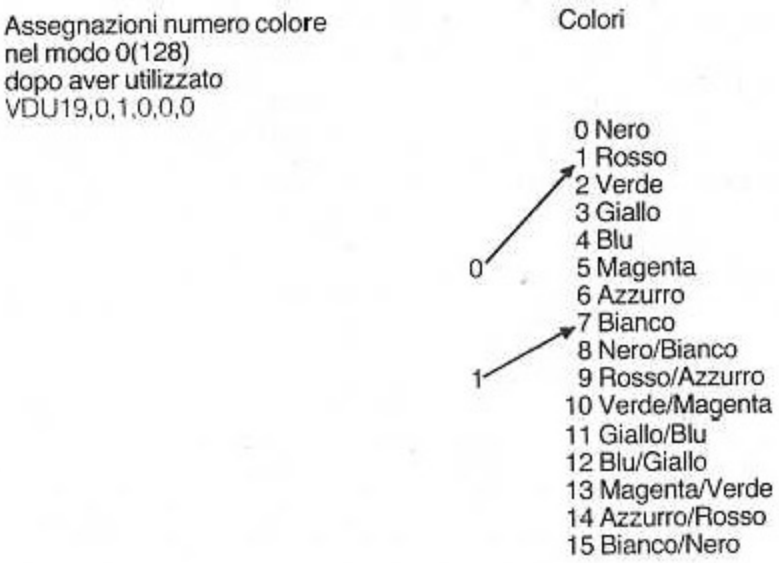
Assegnazioni numero colore nella modalità 0(128)	Colori
	0 Nero
	1 Rosso
	2 Verde
	3 Giallo
	4 Blu
0	5 Magenta
	6 Azzurro
	7 Bianco
	8 Nero/Bianco
1	9 Rosso/Azzurro
	10 Verde/Magenta
	11 Giallo/Blu
	12 Blu/Giallo
	13 Magenta/Verde
	14 Azzurro/Rosso
	15 Bianco/Nero

La lista di destra mostra i numeri dei 16 colori puri. Questa lista non viene mai modificata e vale per tutti i modi. Il modo in cui i numeri di colore per il modo vengono associati alla lista di colori corrente può essere modificato dal comando VDU 19. Provare a digitare:

```
MODE 128 RETURN
VDU 19,0,1,0,0,0 RETURN
```

Queste istruzioni modificano il colore dello schermo da nero a rosso.

Il primo numero dopo VDU 19 è 0, che generalmente produce il colore nero in modo 128. Il secondo numero si riferisce al numero di colore corrente 1 (rosso). Il comando VDU 19 modifica la relazione tra i numeri di colore ed i colori correnti:



Allo stesso modo, per modificare il colore da bianco a giallo, digitare:

VDU 19,1,3,0,0,0 RETURN

(Gli ultimi tre 0 sono previsti per ampliamenti futuri, e devono essere inseriti anche se al momento non hanno alcun effetto).

Lo stesso principio può essere applicato a tutti gli altri modi ad eccezione del 7 e del 135.

Modalità teletext

I modi 7 e 135 visualizzano testo ed immagini in un modo particolare. I comandi quali COLOUR, GCOL, MOVE e DRAW non hanno effetto in questi due modi. Al contrario con i modi 7 e 135 è possibile ottenere visualizzazioni a più colori prodotte per mezzo dei **codici di controllo teletext**.

Il computer permette di produrre proprie visualizzazioni teletext tramite il modo 7 o il modo 135. Questi modi occupano poco spazio di memoria e mettono a disposizione dell'utente una vasta gamma di colori per la visualizzazione simultanea sullo schermo. Le funzioni grafiche sono limitate ma di grande effetto. Nel corso di questa sezione verrà utilizzato solo il modo 135, ma tutte le spiegazioni sono valide anche per il modo 7.

Il programma seguente mostra i colori di testo disponibili in modo 135:

```
10 MODE 135
20 PRINT "Questo";CHR$(129);"mostra che un codice di controllo"
30 PRINT "ha effetto solo sui";CHR$(130);"caratteri"
40 PRINT "che lo seguono sulla";CHR$(131);"stessa";CHR$(129);"riga."
```

L'istruzione PRINT nella riga 20 stampa del testo che contiene una serie di codici di controllo invisibili. Ciascun codice occupa una posizione di carattere, in modo che venga introdotto uno spazio tra una parola e l'altra. I codici influenzano anche il modo in cui vengono visualizzati i caratteri restanti di quella riga. Ad esempio, visualizzando CHR$(129) prima di mostra provoca la visualizzazione del testo in colore rosso, CHR$(130) provoca la visualizzazione del testo in verde, ecc.

La stampa di un codice ASCII tra 129 e 135 influenza il colore dei caratteri visualizzati dopo il codice sulla stessa riga. Digitare:

PRINT CHR$(130) "Prova" RETURN

Questa riga provocherà la visualizzazione del testo in colore verde. Per avere una lista completa dei codici di controllo teletext consultare l'Appendice 3.

Il colore del testo può essere modificato direttamente da tastiera. Tenere premuto SHIFT e premere contemporaneamente il tasto funzione rosso f1. In questo modo verrà visualizzato il codice di controllo 129. Tutti i caratteri battuti sulla stessa riga da questo momento in poi saranno visualizzati in rosso. La pressione di SHIFT e di un tasto funzione da f1 a f7 fornisce un colore diverso per il testo visualizzato successivamente sulla stessa riga.

E' inoltre possibile fare in modo che il testo lampeggi sullo schermo. Digitare:

PRINT CHR$(136);"Lampeggia";CHR$(137);"non lampeggia"; CHR$(136);"lampeggia" RETURN

Per visualizzare il testo a colori e lampeggiante, occorre utilizzare due codici di controllo:

PRINT "Lampeggia";CHR$(129);CHR$(136);"rosso" RETURN

Ciascun codice occupa una posizione di carattere, quindi le parole vengono stampate separate da due spazi.

Anche in questo caso gli stessi effetti possono essere ottenuti con i tasti funzione. Per ottenere il codice di lampeggiamento utilizzare i tasti SHIFT e F8, mentre per il codice di non lampeggiamento utilizzare SHIFT e F9.

Per stampare i caratteri con doppia altezza viene utilizzato CHR$(141):

```
10 MODE 135
20 PRINT CHR$(141);"Doppia altezza"
30 PRINT CHR$(141);"Doppia altezza"
```

Lo stesso testo deve essere stampato su due righe adiacenti a partire da CHR$(141), in caso contrario viene visualizzata solo la parte superiore delle lettere.

Per modificare il colore di sfondo vengono utilizzati due codici:

PRINT CHR$(131);CHR$(157) RETURN

Il primo codice imposta il colore giallo per il testo. CHR$(157) ordina al computer di utilizzare il codice di controllo precedente per il colore di sfondo. L'effetto dei due codici è di produrre un testo giallo su sfondo giallo, come accade eseguendo la riga seguente:

PRINT CHR$(131);CHR$(157);"Ciao" RETURN

Questa operazione è perfettamente inutile in quanto il testo è illeggibile. Per visualizzare un testo su uno sfondo colorato occorrono tre codici di controllo: due codici per modificare il colore di sfondo ed un terzo codice per modificare il colore del testo:

PRINT CHR$(131);CHR$(157);CHR$(132);"Blu su giallo" RETURN

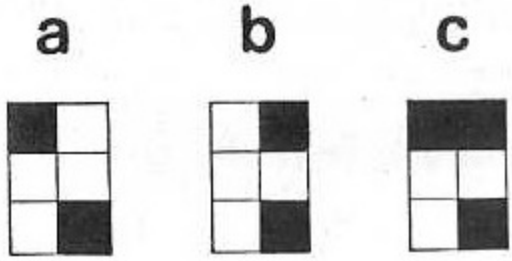
I primi due codici impostano il colore di sfondo su giallo e CHR$(132) è il codice per impostare il testo su blu.

Tutti questi codici possono essere combinati ed incorporati all'interno di stringhe. Per poter utilizzare un set di codici particolare più volte nello stesso programma occorrerà impostare una singola stringa contenente i codici:

```
10 MODE 135
20 rilampeggia$=CHR$(131)+CHR$(157)+CHR$(129)+CHR$(136)
30 PRINT rilampeggia$;"Dimostrazione"
40 PRINT "di visualizzazione normale";rilampeggia$;"e a colori"
```

Funzioni grafiche teletext

Tutte le immagini in modo 135 vengono prodotte come risultato della stampa dei caratteri. Se una riga contiene un codice di controllo grafico, tutti i caratteri all'infuori delle lettere maiuscole che compaiono dopo questo codice sulla stessa riga verranno stampati sotto forma di figure. Ogni lettera corrisponde ad una forma particolare sulla base di una griglia 2x3. Ad esempio:



Nell'Appendice 3 viene fornita una tabella che mostra le forme grafiche associate ad ogni carattere e fornisce i codici di controllo grafici.

La stampa di un codice ASCII da 145 a 151 provoca la stampa dei caratteri sulla stessa riga sotto forma di immagine. Tutte le lettere maiuscole non vengono influenzate:

PRINT CHR$(145);"Aa";CHR$(146);"Bb";CHR$(147);"Cc" RETURN

Si potrà apprezzare la potenza delle funzioni grafiche teletext visualizzando contemporaneamente una serie di caratteri grafici:

PRINT CHR$(148);STRING$(30,"9") RETURN

I caratteri grafici possono essere visualizzati in doppia altezza, su diversi sfondi o lampeggianti.

E' possibile produrre i codici di controllo grafici direttamente da tastiera premendo CTRL contemporanemante a uno dei tasti funzione da f1 a f7. Tutti i caratteri successivi non maiuscoli battuti sulla stessa riga verranno visualizzati con la forma grafica corrispondente.

Suono

Il PC 128 S contiene un **generatore di suono** a quattro **canali**.

Per il controllo del suono sono disponibili due comandi BASIC. Il comando SOUND viene utilizzato per suonare note singole. Ad esempio:

SOUND 1,–15,53,20 RETURN

suonerà una nota sul canale 1 al volume massimo, per la durata di 1 secondo. Il comando può essere riassunto come segue:

SOUND canale,volume,altezza,durata

Il primo dei quattro parametri dopo SOUND indica il numero di canale, che può essere un numero da 0 a 3. Il canale 0 produce rumori per effetti speciali, mentre i canali da 1 a 3 producono note musicali. Digitare:

SOUND 0,–15,53,20 RETURN

Questa riga di programma modifica soltanto il numero di canale impostato nell'esempio precedente, con un risultato completamente diverso.

Il secondo parametro comanda il volume o ampiezza della nota e può essere un valore compreso nella gamma da –15 a 16. Il suono più alto è prodotto da –15, –14 provoca un suono meno alto ed i numeri dal –14 in poi producono suoni sempre più bassi, fino a raggiungere il valore 0 (=silenzio). I numeri positivi da 1 a 16 indicano che il suono è sotto il controllo di un comando ENVELOPE.

Il terzo numero imposta l'altezza della nota e può essere un valore da 0 a 255. I valori più bassi producono note basse, mentre i valori più alti producono note più acute. Il valore di altezza avrà un effetto diverso se viene utilizzato il canale numero 0, cioè il canale dei rumori. In questo caso la gamma di valori per il terzo parametro è da 0 a 7 e produce rumore a diverse altezze.

L'ultimo parametro indica la durata del suono espressa in ventesimi di secondo e può essere un valore compreso tra 0 e 255. Nell'esempio riportato di seguito il valore della durata è 20, quindi la nota verrà suonata per 1 secondo (20 ventesimi di secondo). Il valore 255 produrrà un suono continuo che potrà essere arrestato premendo ESCAPE.

Per poter suonare un semplice motivo sarà sufficiente suonare alcune note in successione:

```
10 SOUND 1,-15,97,10
20 SOUND 1,-15,105,10
30 SOUND 1,-15,89,10
40 SOUND 1,-15,41,10
50 SOUND 1,-15,69,20
```

Per poter suonare le note simultaneamente su un altro canale, aggiungere le righe seguenti:

```
15 SOUND 2,-15,97,10
25 SOUND 2,-15,105,10
35 SOUND 2,-15,89,10
45 SOUND 2,-15,41,10
55 SOUND 2,-15,69,20
```

I suoni con parametri di volume da 1 a 16 sono controllati dall'inviluppo con il numero corrispondente. L'inviluppo influenza sia l'altezza sia l'ampiezza di una nota. Ad esempio:

SOUND 1,-15,255,255 RETURN

suonerà una nota continua ad alto volume. Modificare in 1 il secondo parametro. La nota passerà sotto il controllo dell'inviluppo 1. Il comando ENVELOPE richiede 14 parametri:

ENVELOPE 1,1,-26,-36,-45,255,255,255,127,0,0,-127,126,0 RETURN

Il primo numero che segue ENVELOPE è il numero di inviluppo che può essere un valore tra 1 e 16. I parametri seguenti controllano e modificano l'altezza e l'ampiezza della nota. Provare la stessa nota vista qui sopra, sotto il controllo dell'inviluppo 1:

SOUND 1,1,255,255 RETURN

I comandi SOUND e ENVELOPE sono estremamente versatili e permettono al computer di diventare uno strumento musicale più valido di sintetizzatori molto più costosi. Entrambi questi due comandi sono illustrati dettagliatamente nel manuale di riferimento. Inoltre, nel programma Welcome è compreso un editor di inviluppo (chiamato ENVELOPE) che permette di esercitarsi con i parametri nel comando di inviluppo.

BASIC 128K

Il computer è fornito di una versione su ROM del BASIC BBC che, insieme alle funzioni di memoria shadow accede ad un massimo di 64 Kbyte dei 128 della memoria ad accesso casuale (RAM) disponibile. L'accesso agli altri 64 Kbyte di RAM è reso possibile grazie ad una versione su disco del BASIC BBC (chiamato BAS128), fornito sul disco Welcome. Consultare l'Appendice 10 per ulteriori informazioni su BAS 128.

Linguaggio Assembler

Sebbene i programmi in linguaggio BASIC BBC vengano eseguiti molto velocemente, alcuni programmi, come ad esempio i giochi, devono essere eseguiti ancora più velocemente per poter essere efficaci. Ogni volta che il computer esegue un programma scritto in linguaggio BASIC, deve tradurre (o **interpretare**) ogni istruzioni in modo da eseguire la funzione necessaria utilizzando routine scritte nel linguaggio interno del computer, il *linguaggio macchina*. E' questa operazione di traduzione che rallenta l'esecuzione del programma.

La scrittura di un programma direttamente in linguaggio macchina evita che il computer interpreti ciascuna istruzione, quindi il programma in linguaggio macchina viene eseguito molto più velocemente rispetto all'equivalente in linguaggio BASIC. La scrittura di un programma sotto forma di serie di numeri è però estremamente difficoltosa. Per questo motivo il programma viene scritto in *linguaggio Assembler*.

Il computer traduce un programma in linguaggio Assembler in linguaggio macchina per mezzo di un programma incorporato chiamato **assemblatore**. La traduzione in codice macchina del programma può essere salvata separatamente. Quando il computer successivamente eseguirà il programma, non dovrà tradurre tutte le istruzioni, e l'esecuzione sarà molto più veloce.

Il linguaggio Assembler è più difficile del BASIC, ma crea programmi più corti e più veloci da eseguire. Fortunatamente il computer permette di mescolare i linguaggi BASIC e Assembler in un unico programma e permette di utilizzare esclusivamente il linguaggio Assembler in quelle parti di programma che necessitano un'esecuzione rapida.

Il breve programma riportato di seguito illustra l'uso del linguaggio Assembler. Assicurarsi di digitarlo *correttamente*, come mostrato qui sotto:

```
10 MODE 129
20 DIM demo 10
30 P%=demo
40 [
50 LDA #67
60 JSR &FFEE
70 RTS
80 ]
90 P%=demo
100 END
```

La riga 20 riserva 10 locazioni di memoria per la memorizzazione della versione in linguaggio macchina del programma. *P%* nella riga 30 viene utilizzato per indicare al computer la prima locazione di memoria da utilizzare per il programma in linguaggio macchina.

Le parentesi quadre nelle righe 40 e 80 contrassegnano l'inizio e la fine della parte di programma in linguaggio Assembler. Il breve programma dalla riga 50 alla riga 70 stampa sullo schermo la lettera C. Eseguendo il programma verrà visualizzato quanto segue:

```
E70
E70 A9 43          LDA #67
E72 20 EE FF
                   JSR &FFEE
E75 60             RTS
```

Il computer in questo programma ha utilizzato l'assemblatore per tradurre le istruzioni in linguaggio Assembler in codice macchina. I numeri nella colonna all'estrema sinistra rappresentano le locazioni di memoria esadecimali in cui è memorizzato il linguaggio macchina. Tutti i restanti numeri esadecimali sulla riga sono l'equivalente di un'istruzione in linguaggio Assembler.

Notare che l'esecuzione del programma ha provocato solamente la traduzione del linguaggio Assembler in linguaggio macchina, senza eseguire il programma stesso in linguaggio macchina. Per eseguire effettivamente il linguaggio macchina, digitare:

```
CALL P%
```

CALL ordina al computer di eseguire una parte di programma in linguaggio macchina. Questa istruzione è seguita dalla locazione di memoria da cui dovrà iniziare l'esecuzione del programma in linguaggio macchina. Il computer stampa la lettera C.

Dopo la traduzione di un programma in linguaggio Assembler da parte dell'assemblatore, il programma in linguaggio macchina potrà essere eseguito indipendentemente. Utilizzando NEW, il programma originale viene cancellato, ma il programma in linguaggio macchina rimarrà nella memoria, come può essere verificato battendo CALL P%. La parte di memoria contenente il programma in linguaggio macchina può essere salvata da sola ed utilizzata successivamente senza bisogno del programma originale contenente le istruzioni per il programma in Assembler.

La funzione che permette di mescolare le istruzioni di BASIC e Assembler è molto complessa. Per questo motivo non verrà trattata in questa sede in quanto il presente manuale è stato concepito al solo scopo di fornire nozioni di base per l'uso del computer.

4. Introduzione a VIEW

Che cos'è un word processor

Il word processor ha avuto un impatto molto più forte nei lavori di ufficio di qualsiasi altra applicazione informatica. Pensiamo soltanto a quanti documenti vengono prodotti al giorno in un ufficio o in casa. Lettere, promemoria, elenchi di soci, elenchi di appuntamenti, resoconti... la lista è infinita.

In molti casi il documento dovrà passare diverse revisioni prima della versione definitiva stampata o scritta. Con l'uso di una normale macchina per scrivere, ad esempio, occorrerà produrre una prima bozza, che verrà corretta a mano e quindi ribattuta per ottenere la versione finale. Se successivamente capita di dover riscrivere una lettera uguale, con differenze minime, occorrerà ribattere l'intero documento.

Un word processor offre molti più vantaggi rispetto ad una macchina per scrivere. Il testo battuto con un word processor, prima di essere definitivamente stampato su carta, viene visualizzato sullo schermo per poter effettuare modifiche o per poterlo rileggere. L'esecuzione di correzioni e modifiche è un'operazione molto semplice. E' possibile introdurre, cancellare o spostare in altre parti del documento caratteri, righe o paragrafi interi. Inoltre, il testo può essere memorizzato per poter essere utilizzato ancora successivamente apportando la semplice modifica del nome del destinatario e dell'indirizzo. Si possono produrre lettere personalizzate, tutte uguali ma con nominativi ed indirizzi diversi. Il corpo della lettera deve essere battuto una sola volta.

Il Word processor VIEW

Insieme al computer viene fornito **VIEW**, un potente word processor. Sia per applicazioni di lavoro o hobbistiche, VIEW permette di risparmiare tempo e sforzo nella produzione di qualsiasi tipo di testo.

Prima di iniziare ad utilizzare VIEW, posizionare la mascherina dei tasti funzione sulla parte superiore della tastiera, allineando FORMATTA PARAGRAFO con il tasto **f0**.

Una volta sicuri di avere nel drive il disco Welcome vi sono due metodi per entrare nel word processor VIEW:

a) caricare innanzitutto il menu principale Welcome (vedi capitolo 2). Selezionare poi "Applicazioni" e poi "VIEW"

b) una volta che il computer è pronto ad eseguire programmi in BASIC, probabilmente dopo l'accensione, e dopo aver inserito il disco Welcome nel drive, digitare:

```
*MOUNT RETURN
*LIB LIBRARY RETURN
*WORD RETURN
```

Nel caso si stia utilizzando un monitor a colori è possibile, e spesso desiderabile, modificare la gamma di colori del testo. Per poter modificare il colore prima di entrare in VIEW è necessario seguire il metodo b) sopra descritto. Prima di digitare *WORD occorre però digitare il comando VDU 19 seguito dai necessari parametri. Per informazioni sul comando VDU 19 fare riferimento alla sezione "Modifica della gamma di colori" del capitolo 3.

Due colori di background molto comuni sono il blu e il rosso.

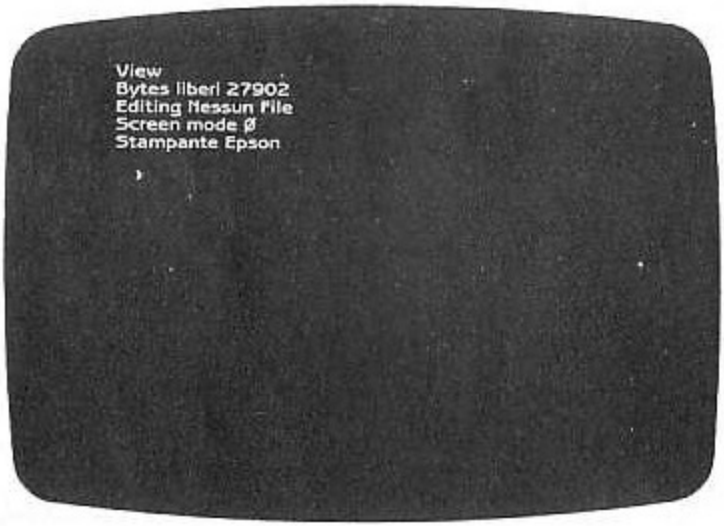
Per avere lo sfondo blu occorre digitare:

VDU 19,0,4,0,0,0

Per avere lo sfondo rosso occorre digitare:

VDU 19,0,1,0,0,0

Una volta entrati in VIEW lo schermo avrà il seguente aspetto:



Se al momento è attivo il modo 7, sullo schermo vi saranno solamente 40 posizioni di caratterè. Per avere 80 caratteri utilizzare il modo 131. Più tardi vedremo come alcune posizioni di carattere su ogni riga sono riservate per scopi particolari. Per selezionare il modo 131, battere:

MODE 131 **RETURN**

Nel corso di questo capitolo si utilizzerà il modo 131. Ricordare che i modi da 128 a 135 sono identici ai modi da 0 a 7 con l'unica eccezione che nei primi la memoria è strutturata per contenere testo. Notare comunque che lo schermo del comando VIEW visualizzerà *sempre* uno dei modi da 0 a 7.

Quello che appare ora è lo **schermo di comando** VIEW, cioè lo schermo da cui vengono impartiti i comandi generali quali ad esempio SAVE. Notare inoltre che i comandi per il sistema operativo (*comando) possono essere immessi da questo schermo. Ad esempio, per accelerare il movimento del cursore si digiterà:

*FX12,3 RETURN

Per tornare alla velocità di cursore normale battere:

*FX12,0 RETURN

Un altro utile comando è *CAT che visualizza un elenco dei file memorizzati o sul disco. Per ulteriori informazioni sui comandi per il sistema operativo vedere l'Appendice 4.

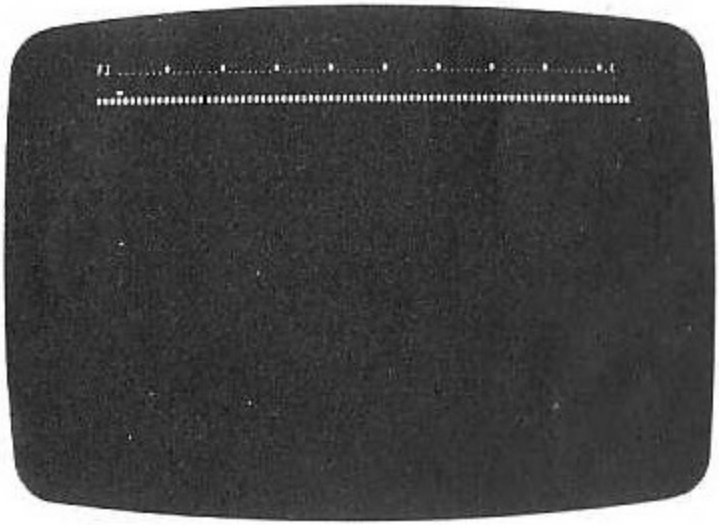
Più avanti in questo capitolo si esaminerà lo schermo di comando. Per il momento per proseguire battere:

NEW RETURN

e premere ESCAPE. In questo modo si attiva lo **schermo di testo.** Il tasto ESCAPE permette di passare dallo schermo di comando allo schermo di testo e viceversa. Notare che il testo introdotto non viene compromesso dalla pressione di ESCAPE.

Introduzione di testo

Lo schermo di testo appare come segue:



Il simbolo lampeggiante è il cursore, alla cui posizione apparirà il testo mano a mano che viene introdotto da tastiera. Battere alcune righe di testo senza premere RETURN. Notare cosa accade quando si raggiunge la fine della riga. Quando ci si avvicina al margine destro del testo ed una parola supera il margine, viene automaticamente portata a capo, all'inizio della riga sottostante. VIEW ha il controllo sulla lunghezza della riga e non divide le parole per portarle a capo.

Tutte le normali funzioni messe a disposizione dalla tastiera sono operative con VIEW esattamente come con BASIC. Ad esempio, se tutto il testo appare in lettere maiuscole, la pressione di CAPS LOCK attiverà la modalità minuscola.

Come vedremo presto, la funzione di portare a capo le parole non è l'unica caratteristica di VIEW. Con VIEW la funzione di **giustificazione** è sempre attiva per default e viene indicata dalla lettera J nell'angolo superiore sinistro dello schermo. Questo significa che il testo viene allineato verticalmente a sinistra e a destra. Per eseguire questa operazione il computer inserisce se necessario degli spazi tra le parole per rendere le righe di uguale lunghezza.

Per disattivare la giustificazione automatica, tenere premuto il tasto CTRL contemporaneamente a f3. Provare ad eseguire questa operazione. Come si può notare la lettera J scompare dalla parte superiore dello schermo. Introducendo a questo punto alcune righe di testo, le parole saranno separate da un solo spazio e le righe saranno di lunghezza diversa, con margine destro a bandiera.

L'attivazione/disattivazione della giustificazione è una delle funzioni contrassegnate sulla mascherina dei tasti funzione. VIEW è stato studiato in modo tale che i comandi utilizzati più frequentemente possono essere ottenuti premendo un tasto funzione. Queste funzioni sono quelle contrassegnate nella riga inferiore della mascherina. La riga mediana della mascherina stessa comprende funzioni ottenibili premendo il tasto SHIFT insieme ad un tasto funzione. Le funzioni contrassegnate nella riga superiore si ottengono tramite la pressione simultanea di CTRL e di un tasto funzione.

Questi comandi sono chiamati comandi immediati in quanto possono essere ottenuti dall'interno dello schermo di testo senza passare allo schermo di comando. Nel presente capitolo si farà riferimento ai comandi immediati tramite il numero di tasto e la sua funzione. Ad esempio:

SHIFT + f7 (IMPOSTA INDICATORE)

significa tenere premuto il tasto SHIFT contemporaneamente al tasto funzione f7.

La riga di punti e asterischi nella parte superiore dello schermo viene chiamata riga campione. La riga campione, oltre ad altre funzioni, indica la lunghezza massima delle righe di testo. Impostando la riga campione è possibile ridurre o aumentare il numero di caratteri che compongono la riga. Premere RETURN due volte per lasciare un po' di spazio e premere:

CTRL + f5 (RIGA CAMPIONE)

Sullo schermo apparirà una copia della riga campione standard. Per mezzo dei tasti con le frecce spostare il cursore all'estremità sinistra della riga campione, quindi cancellare una parte di riga premendo la BARRA SPAZIATRICE per circa dieci volte. A questo punto introdurre un indicatore margine sinistro >. Premere RETURN per iniziare una nuova riga ed introdurre altre due o tre righe di testo e osservare quanto accade dopo aver accorciato la riga campione. L'estremità destra della riga campione può essere impostata allo stesso modo utilizzando però l'indicatore margine destro<.

Modifica di un file testo

La caratteristica peculiare di un word processor è data dalla funzione di modifica e correzione di un testo già memorizzato. Nel disco Welcome è compreso un documento chiamato GRANT1 sul quale possono essere provate tutte le funzioni di modifica.

Visualizzare lo schermo del comando VIEW, cancellare lo spazio di lavoro battendo:

NEW RETURN

Introdurre il disco Welcome nell'unità a disco e, dopo essere entrati in VIEW, battere dallo schermo di comando:

* DIR WELCOME.APPLIC RETURN
LOAD GRANT1 RETURN

Il comando LOAD provoca la sovrascrittura del nuovo testo su tutto il testo correntemente in memoria.

Per tornare allo schermo di testo premere ESCAPE.

Lo schemo apparirà come segue:

Per concatenare invece GRANT1 al testo al momento in memoria introdurre:

READ GRANT1 RETURN

Sullo schermo è visualizzata la prima parte del documento GRANT1. Tenere premuto il tasto freccia verso il basso ed osservare cosa accade quando il cursore raggiunge la parte inferiore dello schermo. L'area di testo VIEW non è limitata al solo schermo. L'area di testo è una pagina piuttosto ampia di cui viene visualizzata solo una parte. Si potrebbe dire che lo schermo è come una finestra che può essere spostata (per mezzo dei tasti freccia) su qualsiasi parte del documento.

Premere il tasto freccia verso il basso fino a visualizzare la fine del documento. Come si può notare il cursore non andrà oltre l'ultima riga di testo. Per oltrepassare l'ultima riga, ad esempio per iniziare un nuovo paragrafo, premere RETURN per aggiungere nuove righe.

L'utilizzo dei soli tasti di movimento cursore per spostarsi all'interno di un documento lungo potrebbe risultare alquanto noioso. Per spostare il cursore sulla parte precedente del testo tenere premuto SHIFT e premere il tasto freccia verso l'alto. In questo modo il cursore si sposta di uno schermo alla volta piuttosto che di una riga per volta. Questa operazione è molto utile per scorrere velocemente il documento. Altre funzioni di spostamento nel documento sono ottenute tramite i tasti funzione f1 (INIZIO TESTO) e f2 (FINE TESTO).

GRANT1 contiene diversi errori, ciascuno dei quali può essere corretto con facilità grazie a VIEW. Per prima cosa si potrà cambiare la data alla lettera. La stessa operazione eseguita con una macchina per scrivere richiederebbe l'uso del correttore liquido, con risultati non completamente soddisfacenti. Se la lettera infatti contiene più di uno o due errori, la procedura consigliabile sarebbe quella di ribattere tutta la lettera. VIEW al contrario permette di correggere gli errori velocemente e con risultati ottimi.

Portare il cursore sul primo carattere della data e battere:

5 marzo

La nuova data sostituirà o "**sovrascriverà**" la vecchia data sullo schermo. Rimarranno alcuni caratteri da *26 febbraio* che potranno essere cancellati posizionando il cursore sullo spazio dopo o premendo DELETE.

Il prossimo errore è costituito dalla parola scritta per errore due volte nella seconda riga. Si potrebbero battere degli spazi sopra ad una delle parole. Questo lascerebbe degli spazi vuoti all'interno della riga. Per evitare quanto sopra, posizionare il cursore sulla prima lettera della parola da cancellare e premere:

f9 (CANCELLA CARATTERE)

Il carattere scomparirà ed il testo che si trova a destra del cursore si sposterà all'indietro ricoprendo lo spazio vuoto. Il cursore dovrebbe ora trovarsi sulla seconda lettera. Premere ripetutamente f9 per cancellare tutte le lettere.

Sulla riga 5 si è omessa una lettera nella parola *prgetto*. Posizionare il cursore sulla lettera *g* e premere:

f8 (INSERISCI CARATTERE)

Tutto quello che si trova a destra del cursore compreso il carattere alla posizione del cursore si sposterà verso destra per creare uno spazio supplementare. Battere ora *o* per completare la correzione di questa parola.

Il prossimo errore si verifica verso la fine del paragrafo in cui è stata omessa una riga dopo la parola *importante*. Posizionare il cursore in qualsiasi punto della riga sottostante e premere:

f6 (INSERISCI RIGA)

Tutte le righe di testo al di sotto di questa riga e compresa la riga stessa si sposteranno verso il basso per lasciare lo spazio per l'inserimento di una nuova riga. Spostare il cursore all'estremità sinistra della riga vuota e battere:

quindi, che tutti gli associati contribuiscano allo

L'ultimo errore in questo paragrafo si trova in fondo, dove una riga è stata battuta due volte. Posizionare il cursore sull'ultima riga e premere:

f7 (CANCELLA RIGA)

Dopo queste correzioni il paragrafo non avrà più l'impostazione corretta. Gli inserimenti e le cancellature hanno provocato un allungamento o un accorciamento di alcune righe, compromettendo l'allineamento a destra. Per rimediare, posizionare il cursore sulla prima riga del paragrafo e premere:

f0 (FORMATTA PARAGRAFO)

Con questa operazione tutto il testo a partire dalla riga in cui compare il cursore fino alla fine del paragrafo verrà riformattato. Attenzione: la modalità di *giustificazione* dovrà essere attiva, stato evidenziato dalla lettera J nella parte superiore dello schermo. Se, dopo l'operazione di giustificazione il paragrafo dovesse rimanere non giustificato, premere:

CTRL + **f3** (GIUSTIFICAZIONE)

ed eseguire ancora la formattazione.

Dopo aver corretto il primo paragrafo si potranno correggere facilmente gli errori presenti nel rimanente testo di GRANT1. L'eventuale formattazione dei blocchi di testo può essere effettuata sia dopo ogni singola correzione sia dopo la correzione di un intero paragrafo. Il risultato finale sarà lo stesso.

La nuova versione di GRANT1 apparirà come segue:

Associazione per la conservazione
Edifici storici

G. Guerra
Via Verdi, 4
20100 MILANO

5 marzo

Caro socio,

in seguito al nostro incontro annuale in gennaio, ho scritto alla OCP per richiedere un aiuto finanziario per la ristrutturazione dell'edificio in Via Muratori. Spero che i nostri associati considerino questo progetto tra i più ambiziosi di quelli intrapresi dalla nostra associazione. E' particolarmente importante quindi, che tutti gli associati contribuiscano allo sforzo richiesto affinché il progetto possa essere realizzato. Ho ricevuto una risposta dal Sig. Stricker della OCP di cui riporto un sunto qui di seguito.

"La OCP non sovvenziona di norma opere di restauro su edifici che avranno un utilizzo commerciale. Siamo comunque consapevoli del fatto che se questo edificio non fosse restaurato si perderebbe una costruzione di grande interesse storico. Di conseguenza, verrà presa in considerazione da parte nostra la possibilità di contribuire a questo progetto".

Si può dedurre che il Sig. Stricker sia ben disposto alla nostra causa e proporrei di accoglierlo presto tra i nostri associati. Gradirei ricevere suggerimenti da tutti gli associati in merito alla forma che deve assumere l'iscrizione di un tale sostenitore. Qualsiasi suggerimento dovrà essermi inviato prima della fine di marzo, in tempo per poterlo presentare all'incontro fissato per il 6 di aprile. Confidando in una tua sollecita risposta, colgo l'occasione per salutarti cordialmente.

Paolo Rossi
Segretario

Dopo aver verificato che tutti gli errori sono stati corretti, occorrerà salvare il documento su disco. Sarebbe consigliabile, anche se si intende proseguire con altre modifiche, salvare di tanto in tanto il documento. In questo modo, in caso di perdita del documento dalla memoria (ad esempio in seguito a cadute di tensione), si perderanno solo le modifiche apportate dopo l'ultimo salvataggio.

Per salvare il file di testo, decidere per prima cosa quale nome assegnare al file. Visto che il primo documento si chiamava GRANT1 si potrebbe chiamare la seconda versione GRANT2.

Estrarre il disco Welcome dal disk drive e sostituirlo con un disco già formattato (vedi Capitolo 6).su cui è possibile effettuare il salvataggio dei file. Battere:

*MOUNT
*DIR
SAVE GRANT2 RETURN

Per richiamare successivamente il documento sarà sufficiente battere:

LOAD GRANT2 RETURN

Operazioni su blocchi di testo

Le funzioni di modifica utilizzate fino a questo momento, ad eccezione di f0 (FORMATTA PARAGRAFO), hanno effetto su una sola riga di testo per volta. Sono comunque disponibili funzioni che influenzano blocchi interi di testo. Per utilizzare queste tecniche, proviamo ad introdurre la canzone *Ten Green Bottles.*

Premere ESCAPE per tornare allo schermo di comando e cancellare lo spazio di lavoro digitando:

NEW RETURN

Premere nuovamente ESCAPE per visualizzare lo schermo di testo ed introdurre la prima strofa come mostrato di seguito. Notare che in questo caso si dovrà premere RETURN alla fine di ogni riga per iniziare la strofa successiva sulla riga sottostante.

Ten green bottles, hanging on the wall
Ten green bottles, hanging on the wall
And if one green bottle should accidentally fall
There'd be nine green bottles, hanging on the wall

E' possibile produrre velocemente tutta la canzone utilizzando la funzione di copiatura blocco. Per prima cosa occorrerà indicare quale blocco di testo deve essere copiato. Per eseguire questa operazione impostare un indicatore all'inizio ed uno alla fine del blocco desiderato che, nel caso specifico, è l'intera strofa.

Posizionare il cursore sulla T all'inizio della prima riga, quindi premere:

SHIFT + f7 (IMPOSTA INDICATORE)

Nella parte superiore dello schermo appaiono i caratteri MK. Premere:

1 per indicare che si sta impostando la posizione dell'indicatore 1.

Verrà visualizzato un rettangolo bianco per indicare la posizione di questo indicatore.

Spostare ora il cursore sulla riga al di sotto dell'ultimo verso della canzone e premere:

SHIFT + f7 (IMPOSTA INDICATORE)

Premere:

2 per indicare che si sta impostando la posizione dell'indicatore 2.

Viene visualizzato un secondo rettangolo bianco che indica la posizione del secondo indicatore.

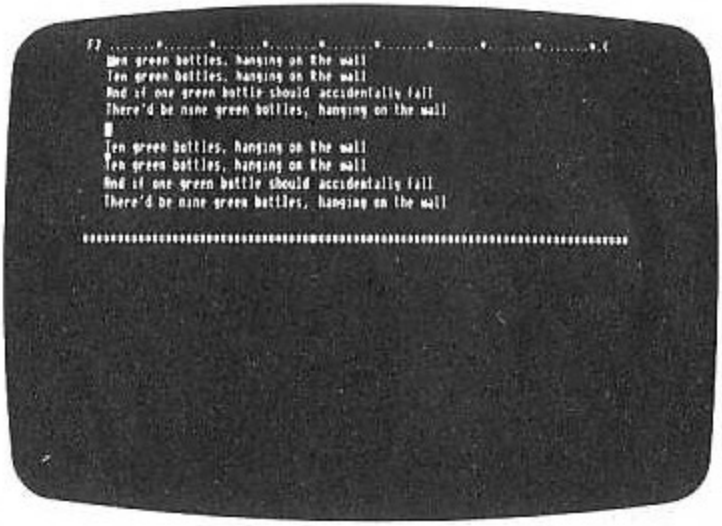
Premendo ESCAPE ed esaminando l'intestazione dello schermo di comando, si noterà la presenza di una riga aggiuntiva che da conferma delle posizioni degli indicatori 1 e 2:

Impostato(i) set 1,2

Premere ESCAPE per tornare allo schermo di testo e spostare il cursore sul punto in cui si desidera far apparire la copia, cioè immediatamente al di sotto del secondo indicatore. Infine, eseguire la copia premendo semplicemente:

COPY

Nel documento saranno così presenti due strofe identiche. Lo schermo dovrebbe apparire come segue:



Notare che i due indicatori sono ancora impostati, quindi, posizionando il cursore e premendo COPY il verso potrà essere riprodotto a piacere. Completare la canzone creando dieci copie del verso, apportando modifiche dove è necessario.

Un'altra utile funzione permette di spostare blocchi di testo da una parte di documento ad un'altra. Impostare gli indicatori per selezionare uno dei versi all'interno del documento *Ten Green Bottles*. Posizionare il cursore in un altro punto del documento e premere:

SHIFT + f0 (SPOSTA BLOCCO)

Il verso contrassegnato verrà trasferito alla posizione del cursore. Si noterà che, al contrario di quanto avviene con COPY, gli indicatori vengono cancellati automaticamente dopo l'operazione di spostamento blocco. La ragione per cui in un'operazione di copia vengono mantenuti gli indicatori è per facilitare l'esecuzione di più copie, come si è visto nella canzone *Ten Green Bottles*.

Un'altra funzione per la gestione di blocchi è la cancellazione, che viene eseguita impostando gli indicatori e premendo:

CTRL + f0 (CANCELLA BLOCCO)

Il testo che si trova al di sotto del blocco cancellato si sposta verso l'alto per coprire lo spazio lasciato dal blocco precedente.

Uso di CHANGE

Supponiamo di avere battuto un documento e di avere commesso lo stesso errore in una parola che si ripete più volte nel testo. Una soluzione potrebbe essere quella di ricercare ogni errore e di correggere le parole una per una. In un documento esteso si dovrà eseguire questa operazione molte volte, con il rischio di dimenticare di correggere degli errori.

Per ovviare al problema viene utilizzata la funzione CHANGE, che possiamo applicare al documento *Ten Green Bottles*.

Dallo schermo di comando, digitare:

CHANGE/green/red/ RETURN

VIEW visualizza il messaggio:

50 stringa(e) cambiata(e)

Ritornando allo schermo di testo ed esaminando il documento si noterà che tutte le parole *green* sono state sostituite con *red*.

In un altro documento si potrebbe voler sostituire *il* con *un* introducendo da tastiera:

CHANGE/il/un/ RETURN

ma il risultato potrebbe non essere quello atteso. Il problema sta nel fatto che VIEW ha trovato ogni *il* da solo, ma anche ogni *il* all'interno di una parola (es. **fil**o, **il**luminazione, oppure **pil**ota). Per risolvere il problema sarà sufficiente battere *il* non da solo, ma introducendo uno spazio prima e dopo questa parola. In questo modo VIEW ricercherà / *il* / e non anche *il* all'interno delle parole.

Provare questa tecnica modificando *un* in *il*, evitando la creazione di parole come **il**ico, oppure pres**il**to. Visualizzare lo schermo di comando e battere:

CHANGE/ un / il / RETURN

Ritornare allo schermo di testo ed osservare il risultato.

CHANGE può essere applicato a frasi o a parole singole. Ad esempio:

CHANGE/ insetto / piccolo animale invertebrato / RETURN

La barra (/) in un comando CHANGE viene denominata **delimitatore** in quanto la sua funzione è quella di contrassegnare l'inizio e la fine di una parola o frase. Nel comando può essere inserito uno spazio al posto della barra purchè non siano richiesti altri spazi all'interno del comando. Ad esempio:

CHANGE visualizzazione schermo RETURN

L'operazione CHANGE è semplicemente una di un gruppo di operazioni **globali** che forniscono funzioni di modifica molto potenti. L'introduzione a tecniche più avanzate non è lo scopo di questo manuale. Gli utenti interessati a queste funzioni dovranno consultare il manuale per l'utente di VIEW.

Ulteriori informazioni sulla riga campione

Cancellare il testo battuto attivando lo schermo di comando e battendo:

NEW RETURN

In base a quanto visualizzato sulla parte superiore dello schermo, controllare se è attivo il modo 131. In caso negativo battere:

MODE 131 RETURN

Premere ESCAPE per attivare lo schermo di testo.

Come si era visto precedentemente, lo stato della riga campione per il testo determina la lunghezza massima di ogni riga di testo. La riga campione nella parte superiore dello schermo corrente è la riga campione standard per il modo 131 e corrisponde ad una lunghezza di riga di 74 caratteri. Ogni modo ha la propria riga campione e quella per il modo 135, ad esempio, corrisponde ad una lunghezza di riga di 34 caratteri.

Per visualizzare un'altra riga campione standard, premere:

CTRL + f5 (RIGA CAMPIONE)

E' buona regola visualizzare sempre una riga campione prima di iniziare l'introduzione del testo per evitare che il testo venga riformattato in seguito se si utilizzerà una riga campione diversa.

Battere ora il testo riportato di seguito. Non è indispensabile premere RETURN alla fine di ogni riga.

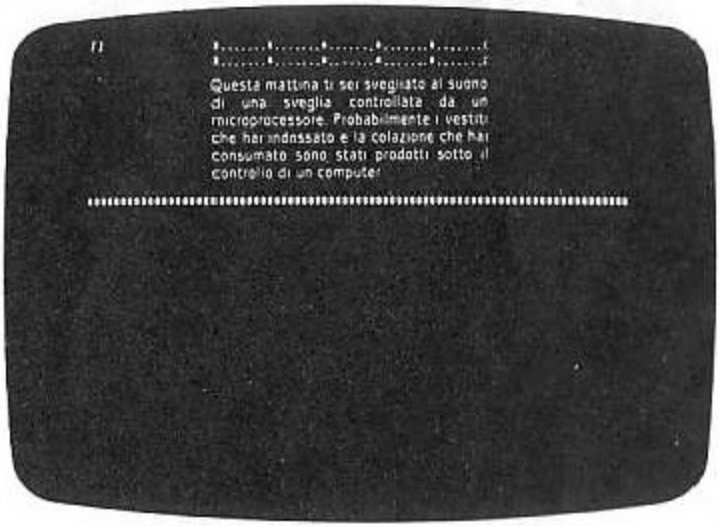
> Questa mattina ti sei svegliato al suono di una sveglia controllata da un microprocessore. Probabilmente i vestiti che hai indossato e la colazione che hai consumato sono stati prodotti sotto il controllo di un computer.

L'impostazione del testo può essere cambiata modificando la riga di campione corrente. Riposizionare il cursore all'interno della riga campione e modificarla fino a renderla uguale alla figura. Per la modifica utilizzare le normali funzioni di editing quali sovrascrittura e cancellazione.

>.....*.....*.....*.....*.....*.....<

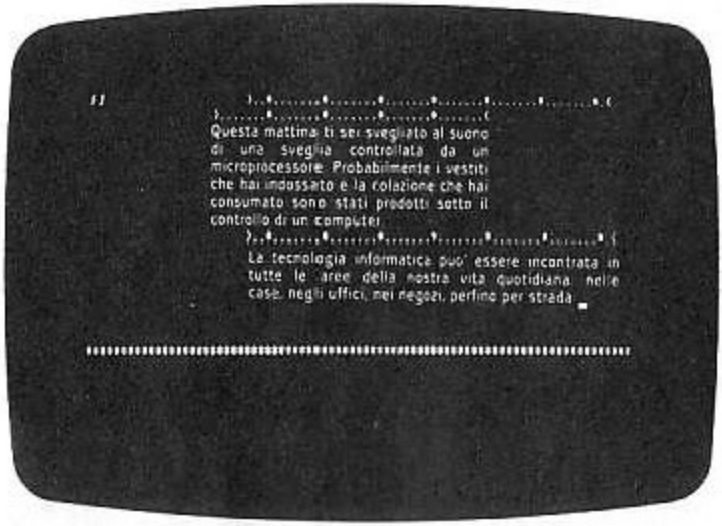
Premere ora RETURN per spostare il cursore dalla riga campione appena modificata alla prima riga del testo. Notare come la riga campione visualizzata nella parte superiore dello schermo sia uguale alla riga campione appena impostata. La riga campione superiore serve a ricordare all'utente quale riga campione è attiva al momento alla posizione corrente del cursore. Premere:

f0 (FORMATTA PARAGRAFO)



Per impostare una riga campione diversa per il paragrafo successivo, sarà necessario aggiungerne al documento una nuova. Per eseguire questa operazione è possibile premere

CTRL + f5 (RIGA CAMPIONE)
per inserire una riga campione standard alla posizione desiderata, e quindi modificando la riga campione in base alle necessità. A volte potrebbe risultare più comodo copiare la riga campione corrente e modificarla. La pressione contemporanea dei tasti SHIFT + COPY genera una copia della riga campione corrente alla posizione corrente del cursore. Dopo aver creato la nuova riga campione, tutto il testo introdotto al di sotto di essa verrà impostato in base ai nuovi margini, come mostrato qui di seguito.



VIEW riconosce la riga campione dai due punti sul margine sinistro. Questi due punti generalmente sono seguiti da una riga di punti e di asterischi alle cui estremità sono presenti le posizioni di margine > e <. Sulle righe campione standard la posizione del margine sinistro non viene indicata. Con due eccezioni, i caratteri visualizzati tra le due posizioni di margine della riga campione sono irrilevanti, quindi sarebbe consigliabile utilizzare una riga di punti per rendere immediatamente riconoscibile la riga campione.

L'asterisco nelle righe campione standard rappresenta le posizioni di tabulazione. La sua funzione può essere illustrata chiaramente posizionando il cursore su una riga vuota e premendo due o tre volte il tasto TAB. Il cursore salta da una posizione di tabulazione alla posizione successiva. Questa funzione è particolarmente utile nella creazione di tabelle. Dopo aver premuto il tasto TAB per spostare il cursore sullo schermo, si potrebbe essere sorpresi dall'effetto provocato dall'uso subito dopo del tasto DELETE. Invece di spostare il cursore di una sola posizione di carattere in questo caso, il tasto provoca il salto del cursore da una posizione di tabulazione alla successiva. Questo effetto è meno sorprendente se si pensa che in effetti la tabulazione è un *carattere* invisibile. Per spostare il cursore di un carattere per volta è sempre possibile utilizzare i tasti freccia o il tasto DELETE.

L'altro carattere speciale che può essere utilizzato in una riga campione è *b* che provoca un segnale acustico simile a quello che si verifica quando si sta per raggiungere la fine di una riga con una macchina per scrivere. Ogni volta che si supera un punto in cui è stato inserito un carattere b nella riga campione, viene emesso un segnale acustico.

Ulteriori operazioni su GRANT2

Caricare il documento GRANT2.

Spesso si rende necessario apportare modifiche consistenti alla struttura e all'impostazione di un documento. In questa sezione modificheremo GRANT2 come mostrato di seguito.

Per prima cosa, centriamo il nome dell'associazione sulla pagina. Questa operazione potrebbe essere eseguita inserendo degli spazi prima del nome, operazione che comporta il conteggio dei caratteri per posizionare entrambe le righe al centro della pagina. Inoltre, se la larghezza del testo viene modificata successivamente, la centratura non sarà più precisa e l'operazione dovrà essere ripetuta.

Per centrare con facilità il testo può essere utilizzato uno dei **comandi memorizzabili** messi a disposizione da VIEW. Questi comandi vengono introdotti nel margine di comando a sinistra dell'area di testo e non hanno un effetto immediato. Essi vengono infatti memorizzati fino alla stampa del documento, e solo a questo punto diventano attivi. Per centrare il testo si utilizza il comando CE (=*CEntrare*). Questo comando esegue la centratura in base alla riga campione corrente e, in caso di nuova impostazione della riga campione, il testo viene centrato di conseguenza.

Visualizzare lo schermo di testo e posizionare il cursore sulla prima delle due righe da centrare. Premere:

SHIFT + f8 (EDITING COMANDO)

Il cursore si sposta sul margine sinistro. A questo punto digitare:

CE RETURN

Il comando CE rimane nel margine ed il cursore si riposiziona all'interno dell'area di testo.

Posizionare il cursore sulla seconda riga da centrare e ripetere l'operazione.

Il comando memorizzabile CE non ha un effetto immediato, ma esiste un modo per eseguire una pre–visualizzazione con la quale il testo appare esattamente come verrà stampato. Attivare lo schermo di comando e battere:

SCREEN RETURN

Sullo schermo appare la prima parte del documento, senza riga campione, e con il nome dell'associazione centrato. Per poter pre–visualizzare l'intero schermo, premere e rilasciare SHIFT. Dopo la pre–visualizzazione dell'intero documento, ritornare allo schermo di testo.

Il comando SCREEN è molto utile per controllare prima della stampa l'aspetto del testo e l'effetto delle righe campione e dei comandi memorizzabili.

Per posizionare l'indirizzo nella parte superiore della lettera viene utilizzato un altro comando. Posizionare il cursore sulla prima riga dell'indirizzo e premere:

SHIFT + f8 (EDITING COMANDO)

Digitare:

RJ RETURN

Introdurre lo stesso comando memorizzabile nel margine sinistro delle altre righe dell'indirizzo e della data, come mostrato di seguito.

RJ è il comando di **allineamento** (o giustificazione) **a destra**. Anche per questo comando l'effetto non è immediato, ma se si attiva lo schermo di comando e si batte:

SCREEN **RETURN**

il risultato dovrebbe essere il seguente:

Premere SHIFT fino al termine del documento, quindi ESCAPE per tornare allo schermo di testo.

Più avanti in questo capitolo verranno esaminati altri comandi memorizzabili. Per il momento vedremo come apportare ulteriori modifiche al documento GRANT2.

Sposteremo ora un blocco di testo dal primo paragrafo alla fine della lettera. Per eseguire questa operazione impostare gli indicatori 1 e 2 all'inizio e alla fine del blocco. Contrariamente a quanto si era visto nell'ultima sezione, questo blocco si trova in mezzo al paragrafo, quindi gli indicatori non potranno essere posizionati su una riga vuota. Impostare l'indicatore 1 sulla *S* della quarta riga e l'indicatore 2 sullo spazio dopo l'ultimo carattere del blocco, cioè prima della *E* sulla sesta riga.

Posizionare il cursore sul punto in cui dovrà apparire il blocco. In questo caso, due righe al di sotto dell'ultimo paragrafo. Premere:

SHIFT + f0 (SPOSTA BLOCCO)

Se spostando un blocco, il formato del testo si è modificato, per riformattare il paragrafo interessato posizionare il cursore sulla prima riga del paragrafo e premere

f0 (FORMATTA PARAGRAFO)

La prossima modifica influenzerà il secondo paragrafo, che dovrà essere formattato con una larghezza minore. Per questa operazione si dovrà introdurre e modificare una nuova riga campione al di sopra del paragrafo interessato. Come si è visto in precedenza, le righe campione influenzano esclusivamente il testo che si trova al di sotto di esse fino a quando non viene incontrata una nuova riga campione. In questo caso riformatteremo un solo paragrafo, quindi si dovrà impostare una riga campione prima e una dopo il testo da reimpostare.

Posizionare il cursore al di sopra della prima riga del paragrafo e premere:

SHIFT + COPY

per ottenere una copia della riga campione corrente. Inserire ora un'altra copia della riga campione corrente al di sotto dell'ultima riga del paragrafo. Modificare la prima di queste due righe per ottenere il formato desiderato.

Notare che non si sono impostate posizioni di tabulazione (*) in quanto in questo particolare documento non si rende necessario alcun rientro di testo. Per i documenti nei quali è richiesta la tabulazione, gli asterischi dovranno essere impostati in base alle necessità.

Quello che rimane da fare è posizionare il cursore sulla prima riga del paragrafo e premere **f0** per eseguire la riformattazione.

Un'altra modifica di GRANT2 consiste nel cancellare l'ultima frase di quello che è diventato il terzo paragrafo.

Posizionare il cursore sulla lettera C all'inizio della frase e premere:

f3 (CANCELLA FINE DELLA RIGA)

Le due righe di testo seguenti possono essere cancellate posizionando il cursore e premendo:

f7 (CANCELLA RIGA)

Infine, modifichiamo OCP in OPC per mezzo del comando CHANGE.

Dopo aver così completato la seconda sessione di modifica, il documento potrà essere salvato sul disco. Per mantenere lo stesso nome che appare nella parte superiore dello schermo di comando, digitare:

SAVE **RETURN**

In questo modo si comunica a VIEW che si desidera utilizzare il nome di file visualizzato correntemente.

Notare comunque che il nome di file sullo schermo di comando può essere modificato in qualsiasi momento. Ad esempio, per modificare il nome del file in ALFREDO, battere:

NAME ALFREDO **RETURN**

Non salvare un documento modificato utilizzando lo stesso nome in quanto se si utilizza un sistema a disco, il nuovo file sovrascriverà qualsiasi altro file con lo stesso nome. Questa operazione potrebbe essere accettabile solo nel caso in cui non si desiderano versioni precedenti del documento. Queste versioni possono essere però utili in caso di perdita del documento definitivo. Se questo è il caso, occorrerà utilizzare un sistema di numerazione, come ad esempio GRANT1, GRANT2, GRANT3 ecc. Non introdurre mai spazi nei nomi di file perchè tutto quello che si trova dopo uno spazio viene ignorato dal computer. Quindi, GRANT 1, GRANT 2 e GRANT 3 verranno trattati tutti come GRANT.

Stampa da VIEW

Quando si lancia la stampa, VIEW invia i codici ad un programma driver stampante che controlla il funzionamento della stampante. Il programma driver di default contenuto in VIEW è studiato appositamente per le stampanti Olivetti Prodest (standard Epson) sebbene sia possibile utilizzare anche driver più sofisticati. Per il momento utilizziamo il driver di stampante di default contenuto in VIEW. Gli altri driver di stampante verranno esaminati più avanti.

La stampa può essere eseguita in due modi. Per stampare una copia del file presente al momento nella memoria del computer, è sufficiente attivare lo schermo di comando e digitare:

PRINT **RETURN**

Per stampare invece un file o più file presenti sul disco senza influenzare il testo contenuto al momento in memoria, procedere come segue. Per stampare ad esempio un file denominato GRANT3, battere:

PRINT GRANT3 **RETURN**

Provare a stampare alcune parti del documento GRANT3 per comprendere il funzionamento di questo comando. Ricordare che per eseguire una pre–visualizzazione prima della stampa definitiva su carta occorre utilizzare il comando SCREEN. Questa operazione è necessaria in caso non si abbia una stampante collegata.

Quando si stampa un documento più lungo di una pagina, VIEW assume una lunghezza di pagina di 66 righe e taglia il testo in base a questo valore di default. Per modificare la lunghezza della pagina stampata, introdurre all'inizio del documento un *comando memorizzabile*. Premere:

SHIFT + **f8** (EDITING COMANDO)

Il cursore si sposta sul margine sinistro. Battere:

PL RETURN

Il comando PL rimane nel margine ed il cursore si riposiziona nell'area di testo. A questo punto digitare:

50 RETURN
Questo numero specifica il numero di righe che dovranno comporre la pagina di testo.

Le successive operazioni di stampa o visualizzazione verranno effettuate in base a questo valore.

A volte il computer taglia un testo in punti indesiderati, come ad esempio a metà di una tabella o subito dopo un titolo (con SCREEN si potranno visualizzare questi salti pagina). Per rimediare a questa situazione occorrerà introdurre dei salti pagina forzati per mezzo del comando memorizzabile PE. Questo comando viene introdotto nel margine, nel punto in cui la stampante dovrà passare alla pagina successiva e sostituisce il salto pagina automatico specificato dal comando memorizzabile PL.

Precedentemente si erano esaminati i comandi CE e RJ. A questa lista aggiungiamo ora PL e PE. Questi comandi servono a mostrare il funzionamento in VIEW dei comandi memorizzati e sono la base delle normali impostazioni di stampa. In VIEW sono comunque disponibili molti altri comandi memorizzati, illustrati nell'Appendice 11 e nel manuale per l'utente di VIEW.

Codici di evidenziazione

Il computer dispone di due tasti funzione, 'EVIDENZIAZIONE 1' e 'EVIDENZIAZIONE 2', che permettono di introdurre "effetti speciali" in un testo (sottolineatura e grassetto). Questi due tasti vengono rappresentati rispettivamente da un trattino e da un asterisco invertiti. Attivare EVIDENZIAZIONE 1 prima e dopo il testo da sottolineare. EVIDENZIAZIONE 2 produrrà una doppia battitura del testo dando così un effetto di grassetto. Attivare EVIDENZIAZIONE 2 prima e dopo il testo da evidenziare. E' possibile utilizzare questi codici su un singolo carattere o su diverse pagine di testo. Si possono utilizzare entrambi questi codici contemporaneamente ponendoli all'inizio e alla fine del testo da evidenziare.

Driver di stampante

I programmi di driver di stampante sono necessari a causa della differenza nella codificazione di caratteristiche speciali delle diverse stampanti e servono per tradurre i codici di VIEW e per controllare la stampante. VIEW è fornito di un driver per le stampanti Olivetti Prodest o compatibili EPSON, ma è possibile modificarlo grazie al comando PRINTER.

Battendo PRINTER e premendo RETURN si reimposterà il driver di stampante rendendolo adatto a stampanti con capacità più limitate ed il testo della parte superiore dello schermo di comando verrà modificato di conseguenza. Fare riferimento al manuale utente di VIEW per ulteriori informazioni sui driver di stampante.

Funzioni supplementari di VIEW

Questa parte conclude l'introduzione al word processor VIEW. A questo punto si sarà compreso il significato di word processing e il funzionamento di VIEW. Come si sarà notato, sono state fornite solo le funzioni di base del programma di word processing. Di seguito verranno illustrate alcune delle funzioni più avanzate di VIEW, che sono spiegate in dettaglio nel manuale utente di VIEW disponibile presso tutti i rivenditori.

Macro

La macro è una parte di testo a cui viene assegnato un nome unico e che può essere incorporata più volte nel testo principale semplicemente ripetendo il suo nome nel margine. La macro può anche essere *modificata* in modo che ogni volta che viene richiamata vengano inserite in posizioni particolari parole o frasi diverse, chiamate **parametri**.

Un uso tipico di una macro potrebbe essere la stampa di lettere standard personalizzate, come citato nella parte introduttiva di questo capitolo. La lettera può essere stampata diverse volte e ciascuna copia conterrà un nome ed un indirizzo diverso. Un'altra possibilità potrebbe essere la creazione di documenti composti da parti di testo standard, ad esempio contratti legali, nei quali devono essere inseriti solo piccoli dettagli per personalizzare il documento prima della stampa.

Il programma Schedario, esaminato nel capitolo 2, dispone di una funzione che permette di "esportare" i nomi e gli indirizzi memorizzati in modo che possono essere utilizzati per la preparazione di lettere personalizzate. L'opzione 'Export' nel menu Schedario genera un file contenente macro VIEW che possono essere lette da VIEW.

Modifiche globali

Precedentemente si è utilizzato il comando CHANGE per modificare parole all'interno di un documento. Operazioni simili comprendono SEARCH, che localizza elementi di testo e REPLACE che esegue le stesse azioni di CHANGE ma offre la possibilità di rifiutare la modifica.

Possono essere utilizzati simboli particolari che rendono queste funzioni valide per i caratteri invisibili quali ad esempio tabulazioni e ritorni carrello. Si può inoltre utilizzare un carattere **wildcard**, ad esempio per rilevare una parola ogni volta che compare nel testo, anche se questa contiene eventuali piccoli errori di ortografia.

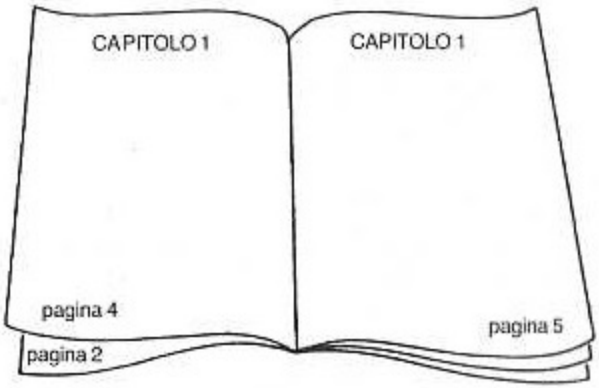
Caratteri accentati

Per visualizzare sullo schermo i caratteri accentati quali à, è, ecc. occorre digitare:

- : seguito da : per ottenere la à
- : seguito da { per ottenere la é
- : seguito da [per ottenere la è
- : seguito da ^ per ottenere la ì
- : seguito da j per ottenere la ò
- : seguito da] per ottenere la ù

Ulteriori funzioni di stampa

Queste funzioni comprendono la possibilità di creare **capi pagina** e **piedi pagina** e possono essere utilizzate per stampare un documento, ad esempio con numerazione automatica di pagina e con il titolo ripetuto su ogni pagina. La figura riportata di seguito mostra un effetto che può essere ottenuto dalla differenziazione di pagine pari e pagine dispari. Fare riferimento al manuale utente di VIEW per ulteriori informazioni.

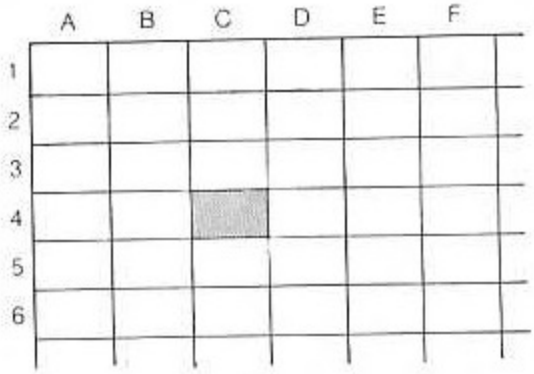


5. Introduzione a ViewSheet

Che cos'è uno spreadsheet

Immaginiamo un grande foglio di carta suddiviso in celle per mezzo di righe orizzontali e verticali. Ciascuna cella può essere identificata individualmente facendo riferimento alla *colonna* e alla *riga*. Le colonne sono contrassegnate A, B, C ecc., mentre le righe sono numerate da 1 a 255. Quindi, si potrebbe fare riferimento ad una **cella** specificando A2 o B4 o F99.

Nella figura riportata qui sotto la cella C4 è stata riempita:



Un programma di spreadsheet simula un foglio di carta come quello nella figura e permette di introdurre in una cella uno dei tre elementi riportati di seguito:

– Un'**etichetta**, cioè una parte di testo
– Un **numero**, cioè 7 oppure 1234.56
– Una **formula** contenente riferimenti ad altre celle

La caratteristica che rende uno spreadsheet uno strumento molto potente è la possibilità di utilizzare formule. Dopo l'impostazione di uno spreadsheet, qualsiasi nuovo valore inserito può essere messo automaticamente in relazione con gli elementi già registrati.

Ad esempio, si potrebbe introdurre il numero 27 nella cella B3 ed il numero 19 nella cella B4. Se si introducesse nella cella B5 la formula B3–B4, il risultato sarebbe quello mostrato di seguito:

	A	B	C	D	E	F
1						
2						
3		27				
4		19				
5		8				
6						

Nelle celle possono essere inserite delle etichette per maggiore chiarezza, come illustrato qui sotto:

	A	B	C	D	E	F
1						
2						
3	VENDITE	27				
4	COSTI	19				
5	GUADAGNI	8				
6						

Il contenuto di ogni cella può essere modificato e gli effetti di tali modifiche influenzeranno anche gli altri valori del foglio. Nel semplice esempio esaminato, la modifica del valore di COSTI in B4 influenzerà automaticamente GUADAGNI in B5.

Chiaramente questo è solo un esempio, ma spiega il valore degli spreadsheet per scoprire cosa succederebbe se si effettuasse una determinata operazione (ad esempio, se prendo in prestito Lit. 1.000.000 per tre anni all'interesse del 13,5% le rate di restituzione saranno di 390.300, ma se aumento le rate a Lit. 450.000....).

I calcoli che sarebbero molto ripetitivi se eseguiti nel modo convenzionale, vengono effettuati semplicemente introducendo i dati su cui i calcoli sono basati. Per questo motivo gli spreadsheet sono molto utilizzati nei settori industria e commercio per previsioni e prospetti finanziari. A casa e nelle piccole aziende risultano utili per i bilanci e la contabilità.

ViewSheet

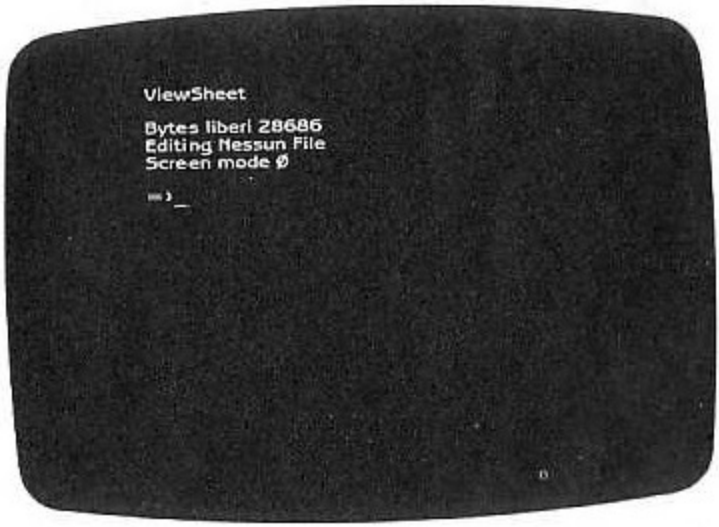
Il PC 128 S Olivetti è fornito di un programma di spreadsheet molto potente che permette di impostare fogli di spreadsheet, di modificarli, salvarli su disco, richiamarli e stamparli per intero o in parte: **ViewSheet**.

Inoltre ViewSheet è compatibile con VIEW. In altre parole, gli spreadsheet possono essere incorporati in testi elaborati con word processor, modificati e stampati come unico file. Se si è letto il capitolo relativo al word processing, si noterà che le operazioni eseguite con VIEW e con ViewSheet sono molto simili.

Uso di ViewSheet

Prima di iniziare ad utilizzare ViewSheet, posizionare la scheda dei tasti funzione nella parte superiore della tastiera, in modo che 'CANCELLA CARATTERE' si trovi in corrispondenza del tasto funzione f9.

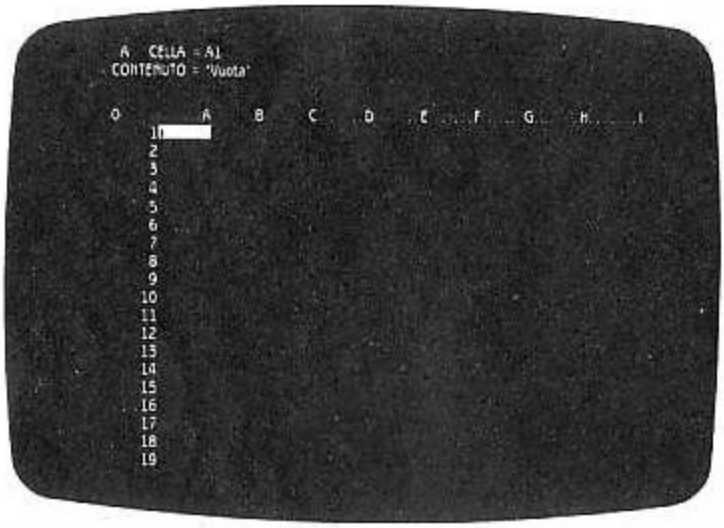
Per avviare ViewSheet, inserire il disco Welcome nel disk drive e premere SHIFT–BREAK. Quando sullo schermo appare il menu principale, selezionare ViewSheet dall'opzione Applicazioni. Dopo alcuni istanti lo schermo apparirà come mostrato di seguito:



ViewSheet sta ora visualizzando lo **schermo di comando**. Premere ESCAPE per attivare lo **schermo di foglio**. La pressione di ESCAPE permette di passare dallo schermo di comando allo schermo di foglio e viceversa. Notare che il contenuto dello spreadsheet non viene influenzato dalla pressione di ESCAPE.

Spostamento all'interno dello spreadsheet

Ora lo schermo appare come mostrato di seguito:



Quello visualizzato è l'angolo superiore sinistro dello spreadsheet. Nella parte superiore saranno presenti le lettere dalla A alla I e sul lato i numeri dall'1 al 19. Quindi, come già detto precedentemente, è possibile fare riferimento ad ogni cella specificando una lettera ed un numero (es. A1 o E15).

Il rettangolo bianco nella cella A1 è il cursore. Una delle funzioni del cursore è di fornire un mezzo per spostarsi all'interno dello spreadsheet. Premere il tasto freccia verso il basso diverse volte per spostare il cursore verso il basso e notare cosa accade quando il cursore raggiunge il fondo dello schermo. La visualizzazione si sposta verso il basso, una riga per volta. Continuando a premere questo tasto si raggiungerà la fine del foglio, cioè la riga 255.

Lo spostamento orizzontale sul foglio avviene allo stesso modo utilizzando i tasti freccia a destra e freccia a sinistra. Le colonne sono contrassegnate dalla A alla Z seguite poi da AA fino a AZ ecc., fino ad arrivare alla colonna contrassegnata IU, che è la 255esima.

Per spostarsi più velocemente nello spreadsheet viene applicata la funzione di *auto–ripetizione*. Premere uno dei tasti freccia e tenerlo premuto per alcuni secondi. Il cursore si sposterà velocemente da cella a cella come se il tasto fosse premuto ripetutamente.

Un altro modo per accelerare il movimento del cursore è tenere premuto il tasto SHIFT e premere il tasto freccia. Invece di spostarsi di una cella per volta, il cursore salta di schermo in schermo. Utilizzare questo metodo per riportare il cursore alla cella A1.

Nel foglio la posizione corrente del cursore viene sempre indicata nella parte superiore dello schermo. A questo punto sullo schermo dovrebbe apparire

CELLA=A1

come posizione corrente, ma questo valore verrà modificato mano a mano che il cursore viene spostato. Nella parte superiore dello schermo viene inoltre visualizzato il contenuto della cella. Al momento dovrebbe apparire quanto segue:

CONTENUTO="Vuota"

Introduzione di informazioni

Assicurarsi che il cursore sia posizionato sulla cella A1, quindi digitare:

NOLEGGIO RETURN

La parola NOLEGGIO appare alla posizione corrente del cursore. Questo foglio utilizza una larghezza di cella di sette caratteri, quindi solo i primi sette della parola NOLEGGIO verranno visualizzati. Spostare il cursore verso il basso nella cella A2 e battere:

IMPOSTA (non premere RETURN)

La parola appare nella parte superiore dello schermo, ma non ancora nello spreadsheet. La parola IMPOSTA si trova nella riga di modifica (editing) e rimarrà in questa posizione finché non si premerà RETURN e quindi nel frattempo potrà essere modificata o cancellata (per mezzo del tasto DELETE) senza influenzare il contenuto del foglio. La riga di editing è particolarmente utile quando si effettuano introduzioni complesse che si vogliono controllare prima di inserirle definitivamente nel foglio. Più avanti in questo capitolo verranno esaminate le funzioni di modifica.

Premere:

RETURN

per trasferire IMPOSTA nel foglio, quindi digitare CARBURANTE nella cella A3 e TOTALE nella cella A5. A4 verrà lasciata vuota per chiarezza di lettura.

NOLEGGIO
IMPOSTA
CARBURANTE

TOTALE

Fino ad ora abbiamo introdotto delle parole o **etichette** nel foglio. Queste etichette non possono essere utilizzate nei calcoli ma sono necessarie come spiegazione dei diversi valori. La lettera L nella parte superiore sinistra dello schermo indica che la cella in cui è posizionato il cursore contiene un'etichetta.

Posizionare il cursore sulla cella B1 e digitare:

126000 RETURN

In questo modo si è introdotto un **valore** nalla cella B1. Un valore può essere un semplice numero o una formula, in ogni caso qualcosa che può essere utilizzato in un calcolo. Mentre il cursore si trova nella cella B1, nella parte superiore dello schermo è visualizzata la lettera V. Questa lettera indica che ViewSheet riconosce il contenuto di B1 come valore e non come etichetta.

Introdurre ora 37000 in B2 e 66000 in B3. Il foglio a questo punto contiene:

NOLEGGIO	126000
IMPOSTA	37000
CARBURANTE	66000

TOTALE

Posizionare il cursore in B5 e battere:

B1+B2+B3 RETURN

Come totale, nella cella B5 apparirà il valore 229000. Notare comunque che la riga CONTENUTO= nella parte superiore dello schermo mostra ancora la formula B1+B2+B3. E' possibile vedere quindi che vi sono due informazioni associate ad una cella contenente una formula: la prima è la formula stessa, in questo caso B1+B2+B3, la seconda è costituita dal numero ottenuto dalla valutazione della formula, in questo caso 229000. Osservando un foglio si può notare che tutti i valori appaiono sotto forma di numeri indipendentemente dal fatto che siano stati introdotti direttamente o calcolati da una formula. La riga CONTENUTO= mostra sempre come si è ottenuto il numero in una particolare cella.

Se a questo punto la cella B5 non contenesse il totale, è probabile che la formula sia stata introdotta in modo errato. Ciascuna delle variabili utilizzate in una formula deve essere un riferimento ad una cella, come ad esempio B1 o AC123. Se ViewSheet non riconosce la formula come riferimento ad una cella, presume che quanto introdotto sia un'etichetta e non la riprodurrà nella cella.

Di seguito vengono forniti alcuni esempi di formule che possono essere utilizzate in ViewSheet. Notare che la moltiplicazione e la divisione vengono eseguite utilizzando rispettivamente i simboli * e /.

Come già detto in precedenza, la caratteristica più importante di ViewSheet è data dalla possibilità di rilevare gli effetti delle modifiche di uno o più valori. Generalmente il modo più veloce per eseguire le modifiche è sovrascrivere il contenuto della cella. Posizionare il cursore in B1 e digitare:

131000 RETURN

Il numero 131000 sovrascriverà il contenuto della cella B1.

Inoltre, il contenuto di B5 verrà modificato e mostrerà il nuovo totale.

A questo punto è possibile provare ad introdurre valori e ad utilizzarli all'interno di formule. L'esempio esaminato precedentemente era particolarmente semplice. Nella prossima sezione si esaminerà uno spreadsheet più realistico e si scopriranno i modi in cui lo spreadsheet può essere impiegato.

Uso di uno spreadsheet

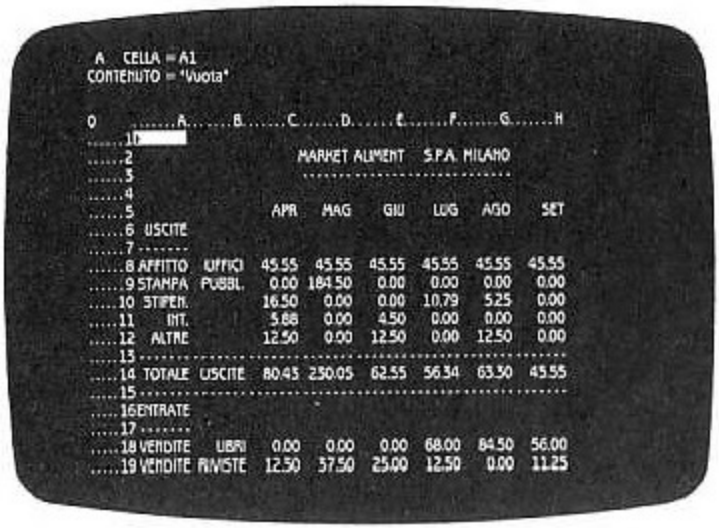
Nel programma Welcome viene fornito un esempio di spreadsheet.

Premere ESCAPE per attivare lo schermo di comando e quindi digitare:

LOAD ACCOUNT RETURN

Il computer caricherà un file chiamato ACCOUNT nello spazio di lavoro di ViewSheet.

Premere ESCAPE per ritornare allo schermo di foglio. Lo schermo avrà l'aspetto seguente:



ACCOUNT mostra il bilancio annuale della MARKET ALIMENT S.p.A. MILANO per l'anno 1985/86. Il foglio completo viene riportato di seguito:

MARKET ALIMENT S.P.A. MILANO — BILANCIO ANNO 1985-86

USCITE		APR	MAG	GIU	LUG	AGO	SET	OTT	NOV	DIC	GEN	FEB	MAR	TOTALE
AFFITTO	UFFICI	45.55	45.55	45.55	45.55	45.55	45.55	45.55	45.55	45.55	45.55	45.55	45.55	546.60
STAMPA	PUBBL.	0.00	184.50	0.00	0.00	0.00	0.00	0.00	99.00	0.00	0.00	0.00	0.00	283.50
STIPEN.		16.50	0.00	0.00	10.79	5.25	0.00	0.00	8.30	6.80	0.00	0.00	0.00	47.64
ALTRE		5.88	0.00	4.50	0.00	0.00	0.00	0.00	0.00	0.00	22.00	3.60	0.00	35.98
INTERE.		12.50	0.00	12.50	0.00	12.50	0.00	12.50	0.00	12.50	0.00	15.00	0.00	77.50
TOTALE	USCITE	80.43	230.05	62.55	56.34	63.30	45.55	58.05	152.85	64.85	67.55	64.15	45.55	991.22
ENTRATE														
VENDITE	LIBRI	0.00	0.00	0.00	68.00	84.50	56.00	40.50	26.00	58.00	0.00	17.50	22.50	373.00
VENDITE	RIVISTE	12.50	37.50	25.00	12.50	0.00	11.25	64.00	80.00	125.00	16.00	0.00	37.50	421.25
VENDITE	VARIE	0.00	650.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	650.00
TASSA	ISCRIZ.	135.00	320.00	200.00	57.00	88.00	275.00	300.00	48.00	25.00	35.00	108.00	321.00	1912.00
TOTALE	ENTRATE	147.50	1007.50	225.00	137.50	172.50	342.25	404.50	154.00	208.00	51.00	125.50	381.00	3356.25
TOTALE	NETTO	67.07	777.45	162.45	81.16	109.20	296.70	346.45	1.15	143.15	-16.55	61.35	335.45	2365.03

La metà superiore del foglio mostra le uscite mensili della società per tutto l'anno con totali mensili nella riga 14. A destra (colonna O) è presente una lista dei totali annuali per ogni articolo. La metà inferiore del foglio mostra una tabella di corrispondenza delle entrate e la riga inferiore (25) dà il totale netto per ciascun mese. Il totale netto dell'intero anno viene fornito nella cella O25.

La maggior parte delle etichette a sinistra del foglio occupano due celle. In questo foglio, tutte le celle sono state impostate ad una lunghezza di 8 caratteri, quindi è stato necessario abbreviare alcune etichette. Questo avrebbe potuto essere evitato inserendo le etichette in una 'finestra' di quindici caratteri di larghezza. Comunque, le tecniche di estensione delle celle e l'impostazione delle finestre non verranno trattate nel presente manuale. Per informazioni consultare la documentazione di ViewSheet.

Inizialmente il cursore è posizionato nella cella A1. Utilizzare il tasto freccia verso il basso per spostare il cursore lungo la colonna A. La lettera L che appare nell'angolo superiore sinistro dello schermo indica che queste celle contengono etichette. Notare che le etichette appaiono nella riga CONTENUTO= nella parte superiore dello schermo.

Posizionare ora il cursore in C8. La lettera L in alto sullo schermo viene sostituita dalla lettera V, ad indicare che la cella contiene un valore.

Spostare il cursore verso il basso per posizionarlo sulla cella C14 ed osservare la riga CONTENUTO=. Questa cella contiene una formula che fornisce i pagamenti totali per il mese di aprile. In effetti C8C12 specifica una **portata** di celle e quando viene usata come formula di cella indica il totale di tutte le celle della portata, cioè C8+C9+C10+C11+C12.

Questo è un esempio di foglio che potrebbe essere utilizzato da una piccola ditta per il calcolo dei salari settimanali ed il totale di questi salari. Questo tipo di utilizzo è particolarmente adatto per uno spreadsheet in quanto implica una grande quantità di calcoli, soprattutto se il personale è numeroso.

Provare ad applicare le tecniche di movimento cursore viste in precedenza. I valori correnti dovrebbero essere gli stessi mostrati nella figura precedente.

Come si può notare, l'etichetta in K2 comprende degli apici. In questo caso è necessario includere delle virgolette, o qualsiasi altro carattere che non può fare parte di una formula, per indicare che 1985–86 deve essere un'etichetta. In caso contrario, ViewSheet sottrarrà 86 da 1985 ed introdurrà il risultato in K2.

Al termine dell'operazione, per ritornare nella parte superiore sinistra del foglio premere:

f7 (VAI A CELLA)

quindi introdurre il riferimento alla cella:

A1 **RETURN**

Questa è una funzione utile per spostarsi velocemente all'interno del foglio, purchè si conosca il riferimento alla cella.

Posizionare il cursore su C12. Questa cella contiene il valore del pagamento dell'affitto in aprile. Modificare il valore in questa cella in 10.25. Quasi immediatamente il valore nella cella C14 viene modificato per fornire il nuovo totale per aprile. Tutti gli altri valori che dipendono da C12 vengono modificati o **ricalcolati**. Per avere una conferma di quanto sopra, provare a rilevare il totale netto in C25 o il totale finale in O25 e confrontare questi valori con quelli originali mostrati nell'ultima figura.

E' importante ricordare il modo in cui ViewSheet esegue il ricalcolo. Il ricalcolo viene effettuato da sinistra a destra, riga per riga dall'alto verso il basso. Quindi se una formula contiene riferimenti a celle che appaiono più avanti nel foglio, potrà non essere ricalcolata quando i valori vengono modificati. In questo caso è necessario premere:

SHIFT + f7 (RICALCOLA)

oppure semplicemente TAB, che avrà lo stesso effetto.

Prima di passare alla creazione di uno spreadhseet occorre ricordare un'altra funzione. In un foglio come ACCOUNT, può capitare di tralasciare una riga o una colonna e questo errore potrebbe essere rilevato solo quando il foglio è completo.

In ACCOUNT, ad esempio, potrebbe essere stata dimenticata una riga di entrate di una concessione comunale. Posizionare il cursore in qualsiasi punto della riga 21 e premere:

SHIFT + f2 (INSERISCI RIGA)

Tutto quello che appare al di sotto del cursore verrà spostato verso il basso per fare posto alla nuova riga. Inserire l'etichetta conces. nella cella A21 e comun. nella cella B21.

Posizionare il cursore nella cella C21 ed introdurre un valore, ad esempio 75. La prima cosa che risulta è che, a differenza degli altri valori del foglio, questo non verrà visualizzato con due posizioni decimali. Per il momento ignorare questo fatto. Più avanti in questo capitolo si vedrà come formattare una cella in modo che possa visualizzare un numero specificato di posizioni decimali.

Un punto interessante da notare è che il valore in C24 è stato ricalcolato per visualizzare il nuovo totale, anche se la formula originale in C23 non comprendeva la nuova riga. Questo perchè ViewSheet ha sistemato automaticamente la portata specificata in C24 per comprendere la nuova riga. Questo può essere verificato posizionando il cursore in C24 ed osservando la riga CONTENUTO=. Originariamente questa riga conteneva C18C21, ora contiene C18C22.

Allo stesso modo, si possono inserire delle colonne premendo:

SHIFT + f1 (INSERISCI COLONNA)

Prima di concludere le operazioni su ACCOUNT, provare ad utilizzare INSERISCI RIGA e INSERISCI COLONNA e le funzioni complementari fornite da:

CTRL + f1 (CANCELLA COLONNA) e
CTRL + f2 (CANCELLA RIGA)

Si noterà che inizialmente sarà impossibile cancellare righe e colonne. Questo perchè è presente una **protezione** che deve essere disattivata. Premere ESCAPE per passare allo schermo di comando e battere:

PROTECT OFF RETURN

Sullo schermo appare il messaggio:

Protezione off

Premere ESCAPE per tornare allo schermo di foglio per cancellare righe e colonne. Sarebbe consigliabile riattivare la protezione dopo aver eseguito la cancellazione desiderata. Per attivare la protezione digitare:

PROTECT ON RETURN

dallo schermo di comando.

Creazione di uno spreadsheet

A questo punto si potrà creare uno spreadsheet da una bozza su carta. L'equivalente di questa bozza viene mostrato di seguito.

Salari settim. settim. 21

ore normali 40 coeffic. straor. 1.75

NOME	ORE TOTALI	ORE NORMALI	ORE STRAOR.	TARIFFA NORMALE	TOTALE
Franco	40	40	0	4.50	180.00
Paolo	34	34	0	4.50	153.00
Bruno	48	40	8	5.00	270.00
Sergio	43	40	3	4.50	203.62
Filippo	38	38	0	5.25	199.50
Carlo	52	40	12	5.25	320.25
TOTALI	255	232	23	28.00	1326.37

Nell'esempio i salari vengono calcolati sulla base di una normale settimana di lavoro di 40 ore. Le ore di straordinario sono calcolate 1,75 volte il salario orario base.

La riga superiore del foglio contiene un titolo ed il numero di settimana. La riga successiva specifica il numero di ore normali e le ore straordinarie.

La parte rimanente del foglio consiste di una tabella con sei colonne. La colonna 1 elenca i nomi degli operai e la colonna 2 elenca le ore lavorate da ciascuna persona. Le colonne 3 e 4 suddividono le ore lavorate in normali (40 o meno di 40) dalle ore di straordinario. La colonna 5 elenca la tariffa oraria per ogni operaio e l'ultima colonna il totale generale. Alla fine di ogni colonna è presente il totale parziale e la parte finale della colonna 6 indica il totale generale.

Introduzione del foglio

Dallo schermo di comando selezionare il modo 131 quindi digitare:

NEW RETURN

per cancellare lo spazio di lavoro.

Premere ESCAPE per attivare lo schermo di foglio.

Il cursore dovrebbe trovarsi nella cella A1 e sarà in questa cella che introdurremo il titolo. Battere:

Salari RETURN

Ricordare di premere CAPS LOCK se tutte le lettere devono essere scritte maiuscole.

Questo foglio utilizza una larghezza di cella di sette caratteri. Sebbene in una cella possano essere introdotti 239 caratteri, solo i primi sette verranno visualizzati. La parte rimanente del titolo dovrà essere introdotta nella cella B1. Posizionare il cursore in B1 e battere:

settim. RETURN

Allo stesso modo introdurre i caratteri di sottolineatura nelle celle A2 e B2.

Battere ora "sett." in E1 e 21 in F1.

Si potrebbe introdurre "*sett. 21*" in un'unica cella, ma verrebbe trattata come un etichetta. Il presente foglio verrà probabilmente utilizzato per tutte le settimane dell'anno, quindi 21 deve essere inserito come valore, in modo che possa essere facilmente modificato.

Completare ora la riga 4 battendo:

ore	in A4
normali	in B4
40	in C4
coeffic	in D4
straor.	in E4
1.75	in F4

Come si sarà notato, l'impostazione delle colonne è piuttosto strana. Le etichette come ore e normali sono allineate a sinistra mentre i valori come 40 e 1.75 sono allineati a destra.

Per aumentare lo spazio sarà necessario spostare i valori in F1, C4 e F4 da destra a sinistra nelle celle rispettive.

Posizionare il cursore nella prima delle tre celle menzionate qui sopra e premere:

f6 (MODIFICA FORMATO CELLA)

Nella parte superiore dello schermo appare:

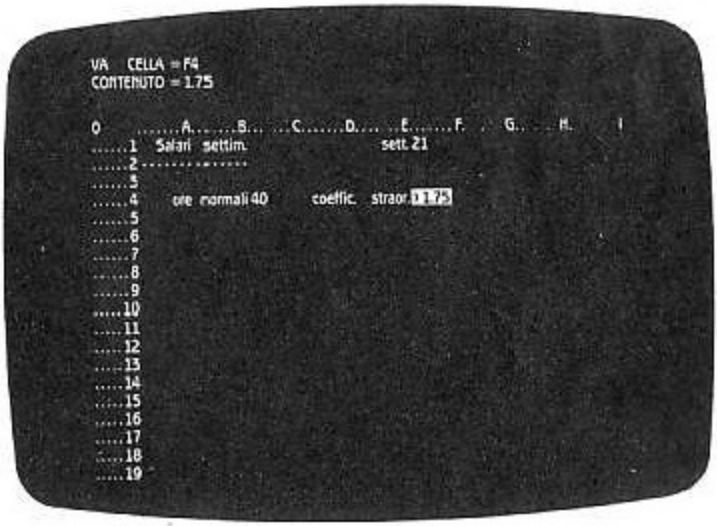
Formato?
FRM

Battere:

L **RETURN**

Il numero verrà allineato a sinistra. Ripetere la procedura per le rimanenti due celle.

Lo schermo dovrebbe apparire come mostrato qui sotto:



Introdurre ora i titoli delle colonne da A a F nelle righe 7 e 8. Per mantenere l'impostazione dello stile utilizzare lettere maiuscole.

Prima di battere la lista di nomi nella colonna A, premere:

CTRL + f0 (INTRODUZIONE AUTOMATICA)

Nella parte superiore sinistra dello schermo appare la lettera R dopo LA. Premere nuovamente la combinazione. La lettera R viene sostituita dalla lettera D. Premere di nuovo per far scomparire la lettera D.

R sta per destra, mentre D sta per basso. INTRODUZIONE AUTOMATICA permette di evitare di dover premere i tasti freccia ogni volta che ci si vuole spostare su una cella adiacente. INTRODUZIONE AUTOMATICA è utile quando si introduce una riga o una colonna di elementi molto lunga. Per spostare il cursore automaticamente sulla cella successiva è sufficiente premere RETURN.

Per introdurre la lista di nomi occorre utilizzare il modo D (basso). Questo serve anche per introdurre le righe in A15 e A17 e l'etichetta TOTALI in A16.

Quando INTRODUZIONE AUTOMATICA è attivato, è possibile utilizzarlo per introdurre la lista ORE TOTALI nella colonna B. Per il momento però non effettuare introduzioni oltre B14.

Disattivare INTRODUZIONE AUTOMATICA premendo:

CTRL + f0 (INTRODUZIONE AUTOMATICA)

Ogni elemento della riga 15 è identico. Non è necessario ribattere l'etichetta '-----' per sei volte in quanto è possibile duplicarla. Premere:

f0 (DUPLICA)

Il sistema emetterà il prompt:

Da – a ?

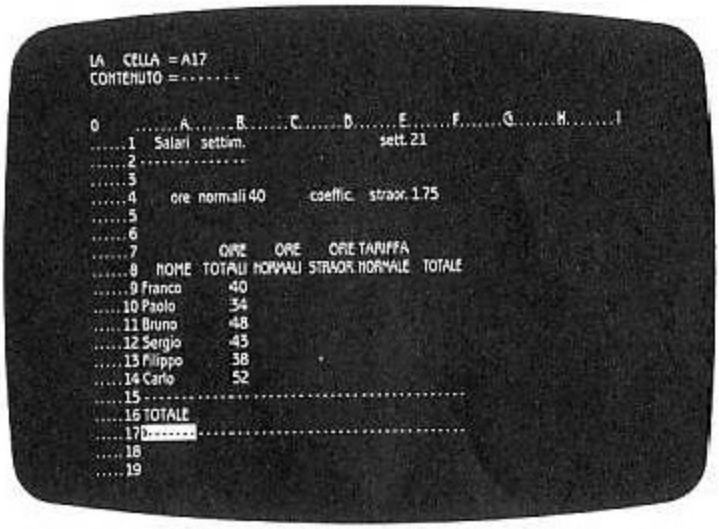
Battere:

A15 – B15F15 RETURN

In questo modo il contenuto di A15 viene **copiato** nelle colonne da B a F nella riga 15. Come si è visto, la specifica di una portata, come B15F15 permette di eseguire diverse operazioni con ViewSheet.

Ripetere la procedura per introdurre la riga 17.

Il foglio dovrebbe ora apparire come segue:



Prima di continuare è importante distinguere tra i valori da introdurre direttamente e quelli che ViewSheet può calcolare da una formula. In base alle regole viste in precedenza, e ad altre che vedremo ORE NORMALI e ORE STRAOR. possono essere calcolate da ORE TOTALI. Allo stesso modo TOTALE verrà calcolato dai valori nelle colonne precedenti. Quindi, gli unici valori da introdurre saranno quelli nella colonna E, cioè TARIFFA NORMALE.

Battere il primo valore, 4.50 in E9. Come si può notare lo zero finale viene ignorato e viene introdotto 4.5. Per somme di denaro è necessario avere un formato di celle che mantenga le due posizioni decimali. Premere:

f6 (MODIFICA FORMATO CELLA)

In risposta a:

Formato?
FRM

Battere:

D2 **RETURN**

che significa due posizioni decimali. Notare che è possibile combinare più istruzioni per la modifica dei formati delle celle battendo ad esempio D2L per specificare due posizioni decimali e giustificazione a sinistra.

Provare ad introdurre gli elementi rimanenti nella colonna E utilizzando INTRODUZIONE AUTOMATICA. Ricordare di modificare ogni formato di cella per impostare le due posizioni decimali.

Formule

A questo punto sono rimaste tre colonne vuote ed una riga TOTALI vuota nella parte inferiore della tabella. I valori in queste posizioni verranno calcolati da ViewSheet. Quello che resta da fare è trovare una formula adatta per ogni cella.

Iniziare con le formule nella riga TOTALI in quanto sono le più semplici. Per ottenere un totale per la colonna B, ORE TOTALI, posizionare il cursore in B16 e battere:

B9+B10+B11+B12+B13+B14 RETURN

Quasi immediatamente la somma dei sei numeri appare in B16. Sebbene questo metodo sia perfettamente accettabile per sommare liste relativamente corte di valori, è poco pratico per somme molto lunghe. Un modo migliore per questo tipo di addizioni è specificare una portata di valori, come mostrato in precedenza nello spreadsheet ACCOUNT. Lasciando il cursore in B16 premere:

SHIFT + f9 (CANCELLA CELLA)

In questo modo il contenuto corrente di B16 viene cancellato. Ora battere:

B9B14 RETURN

Il risultato dovrebbe essere la somma dei valori in tutte le celle da B9 a B14 comprese.

Le portate dovranno essere specificate in un modo simile per le altre celle nella riga 16. Per eseguire ciò, effettuare una duplicazione anche se le formule contengono diversi riferimenti a celle. Premere:

f0 (DUPLICA)

Sullo schermo apparirà:

Da – a ?

Battere:

B16 – C16F16 RETURN

Verrà visualizzato il messaggio:

R)elativa, N)on cambio?
B9 B14

Notare che il riferimento alla cella B9 è evidenziato.

L'operazione di duplicazione vista in precedenza inerente ad una etichetta veniva applicata ad una cella che non conteneva formule.

ViewSheet permette di duplicare il riferimento a celle in modo **assoluto o relativo**.

La duplicazione assoluta copia il contenuto esatto di cella alla portata specificata, esattamente come si era fatto per duplicare un'etichetta. La duplicazione relativa modifica B16 in C16, D16, E16 e F16 in ogni cella successiva. Per selezionare la duplicazione relativa su B9 premere R.

Lo schermo visualizzerà:

R)elativa, N)on cambio?
B14

Ora è possibile selezionare la duplicazione assoluta o relativa per il riferimento alla cella B14. In questo caso non si vuole copiare B14 esattamente come appare, ma si desidera modificarlo in C14, D14, E14 e F14. Premere R.

Ora. Facendo scorrere il cursore si può vedere nella riga CONTENUTO= nella parte superiore dello schermo come la formula è stata copiata ogni volta con il riferimento aggiornato alla cella.

Provare ora ad utilizzare la duplicazione per introdurre delle formule nella colonna D, ORE STRAOR. Lo straordinario è calcolato semplicemente sottraendo ORE NORMALI da ORE TOTALI. Quindi D9 ad esempio, conterrà B9–C9. Introdurre questa formula in D9 quindi utilizzare la duplicazione relativa su B9 e C9 per completare la colonna.

Per completare la colonna C, ORE NORMALI, non è sufficiente una semplice formula. Il valore viene calcolato per ogni operaio osservando ORE TOTALI. Se questo valore è 40 o inferiore, ORE NORMALI avrà la stessa cifra. Se supera 40, ORE NORMALI sarà 40. Per il primo operaio, Franco, si potrà rappresentare questo come:

'Se B9 è maggiore di 40, introdurre 40, altrimenti introdurre B9'

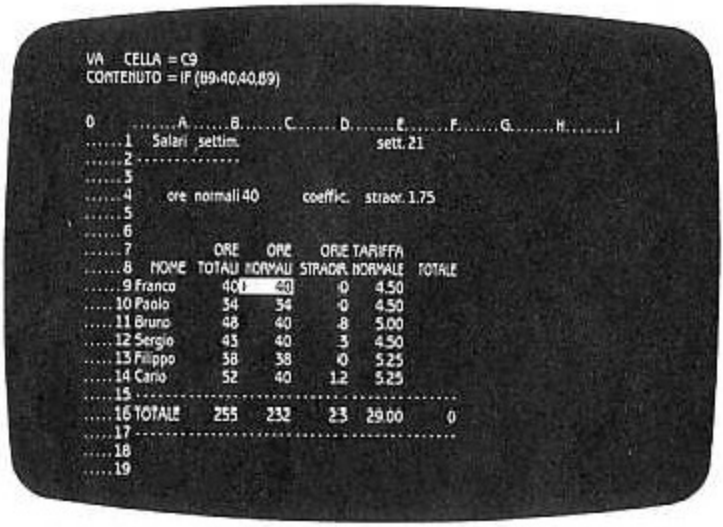
che, tradotto in formula diventerà:

IF(B9>40,40,B9)

Questa è una funzione di spreadsheet molto potente che permette di ottenere dalla condizione (B9>40) il valore (40 o B9) da assegnare ad una cella.

Introdurre la formula precedente nella cella C9 ed utilizzare la duplicazione per inserire la formula corrispondente nella parte rimanente della colonna C.

A questo punto, il foglio dovrebbe apparire come segue:



Tutto quello che rimane è completare l'ultima colonna, TOTALE. Questo valore è composto da due parti, la paga base e la paga straordinaria. Prendendo come esempio Franco, il totale verrebbe calcolato come segue:

paga base
= ORE NORMALI x TARIFFA NORMALE
= 40 x £4.50
= £180

paga straordinaria
= ORE STRAOR. x TARIFFA NORMALE x COEFFIC STRAOR.
= 2 x £4.50 x 1.75
= £15.75

totale
= paga base + paga straordinari
= £195.75

Tradotto in formula si avrà:

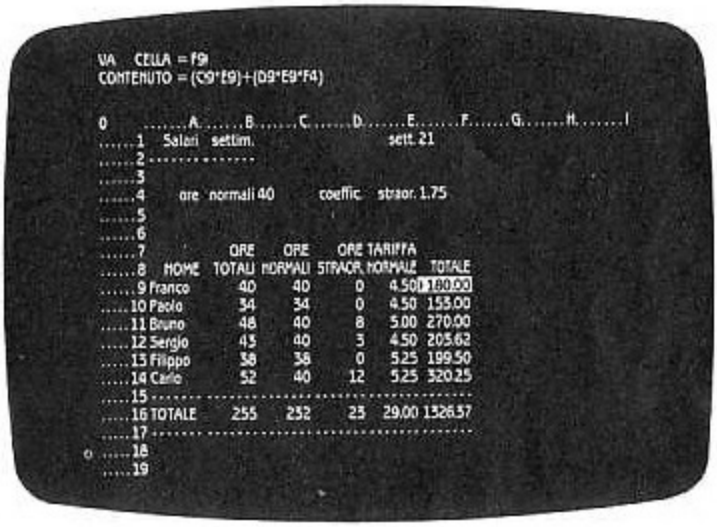
(C9*E9)+(D9*E9*F4)

Introdurre questa formula in F9 quindi duplicarla nelle celle da F10 a F14. Notare che F4 è il coefficiente di straordinario per tutti gli operai e deve essere copiato esattamente per ogni cella. Quindi, quando appare:

R)elativa, N)on cambio?
F4)

Si dovrà rispondere N (*Non cambio*)

Il foglio dovrebbe apparire come mostrato di seguito:



Si potrebbe avere l'impressione che il procedimento di creazione di uno spreadsheet così semplice sia alquanto laborioso, ma i principi adottati sono gli stessi per un foglio che riguarda sei persone ed un foglio che ne riguarda sessanta. Il metodo utilizzato nella creazione del foglio SALARI servirà da guida per la creazione di spreadsheet molto più utili e complessi.

Come già menzionato precedentemente, gli spreadsheet sono particolarmente utili per rispondere a domande del tipo: "Cosa succede se...?". Di seguito vengono forniti tre problemi che risolveremo utilizzando il foglio SALARI e che dimostreranno l'utilità dello spreadsheet nello svolgere un'operazione che, se eseguita con un foglio di carta ed una matita richiederebbe una grande quantità di calcoli.

1 Dopo aver completato il foglio dei salari settimanali ci si rende conto che è stato commesso un errore nel conteggio Sergio e che effettivamente devono essergli pagate solo 35 ore. Modificare il foglio in base a questo dato.

2 La società ha intenzione di aumentare il coefficiente di straordinario da 1.75 a 1.95 e si vuole valutare l'effetto di questo aumento sul totale pagato.

3 Si ha inoltre intenzione di portare il numero di ore settimanali da 40 a 38.

Questo problema mette in evidenza un difetto nello spreadsheet. Sebbene sia stato introdotto 40 come *valore* nella cella C4, tutte le formule relative faranno riferimento alla costante 40 invece che al riferimento alla cella C4. Se nelle formule vengono utilizzati riferimenti a celle, l'intero spreadsheet verrà ricalcolato ogni volta che viene modificato il valore di una singola cella. Quindi, se è possibile scegliere, le formule dovrebbero contenere riferimenti a celle invece che solo valori.

Stampa

La stampa da ViewSheet è molto simile a quella da View in quanto i codici vengono inviati ad un *driver di stampante* che controlla il funzionamento della stampante. Nel PC 128 S è presente un driver incorporato che permette di utilizzare stampanti compatibili Epson. Le stampanti Olivetti Prodest sono compatibili Epson. Se si è in possesso di una stampante non compatibile Epson, sul disco Welcome sono forniti i diversi driver di stampante.

Con il sistema impostato correttamente, la stampa del foglio appena creato sarà molto semplice. Attivare lo schermo di comando e battere:

PRINT **RETURN**

In questo modo viene stampata l'area del foglio visualizzata sullo schermo. In modo 3 questo comprende tutto quello che si trova tra A1 a I19 e nel caso specifico comprende quindi l'intero foglio.

Per la stampa di quanto si trova al di fuori dell'area da A1 a I19, ViewSheet utilizza le **finestre di stampa**. Per informazioni consultare la documentazione di ViewSheet.

Uso di uno spreadsheet con VIEW

E' possibile incorporare spreadsheet nei file VIEW. Questa caratteristica è molto utile per la creazione di documenti che contengono tabelle.

Per salvare il foglio in un formato corretto (ad esempio chiamandolo SOLDI), procedere come segue:

Battere:

*SPOOL SOLDI **RETURN**
SCREEN **RETURN**

Sullo schermo viene visualizzato il foglio. Ora battere:

*SPOOL **RETURN**

Con questa procedura viene salvata la parte di schermo da A1 a I19. Per i fogli che occupano più spazio di questa area devono essere impostate finestre di stampa. Per informazioni consultare la documentazione di ViewSheet.

Dopo aver effettuato il salvataggio, il file può essere letto in VIEW e quindi modificato o incorporato in un altro documento. Notare che i file di spreadsheet creati in questo modo devono essere caricati per mezzo del comando READ di VIEW e non del comando LOAD.

Altre funzioni di ViewSheet

Con un po' di esperienza si potranno apprezzare le caratteristiche di ViewSheet ed utilizzare le potenti funzioni messe a disposizione da questo programma. Siamo stati costretti a tralasciare molte delle funzioni più avanzate di ViewSheet, alcune delle quali vengono trattate in breve di seguito.

L'Appendice 12 fornisce un riassunto dei comandi dello schermo di comando di ViewSheet. Per informazioni sul loro utilizzo consultare la documentazione di ViewSheet.

Quando si lavora su un foglio molto vasto, potrebbe risultare utile visualizzare diverse parti del foglio simultaneamente. Ad esempio, supponiamo di dover sommare una lista di valori nella colonna A. Se non si è sicuri del numero di elementi contenuti nella lista, si potrebbe inserire il totale in fondo alla colonna nella cella A255, ma al termine dell'operazione risulterebbe noioso dover continuare a visualizzare diverse parti del foglio per fare riferimento agli elementi della lista ed al totale in A255.

Questo problema può essere risolto utilizzando le **finestre di schermo** di ViewSheet. A255 può essere visualizzata in una finestra in fondo allo schermo, mentre la lista dei valori nella colonna A viene fatta scorrere indipendentemente.

Possono essere impostate fino a dieci finestre contemporaneamente, ciascuna delle quali può essere di dimensioni diverse. Questo permette di utilizzare celle ampie per le etichette e celle più piccole per i valori.

Come citato in precedenza, le finestre sono utili anche per la stampa. Diverse parti di un foglio possono essere stampate una accanto all'altra oppure incorporate in un documento VIEW.

ViewSheet fornisce diverse **funzioni** da utilizzare nelle formule di cella. Una di queste funzioni è già stata utilizzata: IF. Vi sono funzioni che possono ad esempio fornire i valori minimi, massimi o medi per una portata specifica. Alcune funzioni hanno il corrispondente nel BASIC BBC, ad esempio INT, LOG e SIN.

All'interno di uno spreadsheet può essere impostata una **tavola di look–up**. ViewSheet effettuerà una ricerca in una lista di valori fino a trovare l'elemento specificato. Il valore corrispondente in una seconda tavola verrà quindi localizzato ed utilizzato come richiesto. Ad esempio, supponiamo che la prima lista contenga codici numerici e la seconda prezzi. Si potrebbe fare riferimento ad un codice in un altro punto del foglio e in base a questo visualizzare il prezzo corrispondente.

Un'altra funzione utile permette la visualizzazione dei dati sotto forma di diagrammi a barre. All'interno di una parte specifica di foglio, i numeri vengono rappresentati automaticamente per mezzo di linee o asterischi.

6. Sistemi di archiviazione

Che cos'è un sistema di archiviazione

Virtualmente ogni programma applicativo (a parte quelli più semplici) necessita di accedere ad un dispositivo di memorizzazione esterno, ad esempio un disco magnetico, su cui registrare i dati. Questo per diversi motivi, tra cui il più importante è che il contenuto della memoria del computer non può essere mantenuto dopo lo spegnimento del computer stesso. Un altro motivo è che generalmente i programmi lavorano con una quantità maggiore di dati di quanti ne può contenere la memoria principale del computer oltre al programma.
Ovviamente, se i programmi ed i dati devono essere memorizzati esternamente alla memoria, l'utente dovrà essere in possesso di un mezzo adatto che possa:

- recuperare (caricare) gli elementi esistenti;
- accedere agli elementi esistenti in modo selettivo (cioè senza doverli caricare per intero);
- memorizzare (salvare) nuovi elementi.

Sistemi di archiviazione standard

Il PC 128 S è fornito di due sistemi di archiviazione standard:

- il sistema di archiviazione ROM (RFS)
- il sistema di archiviazione avanzato su disco (ADFS)

che permettono al computer di accedere rispettivamente ai file contenuti nei chip della memoria di sola lettura ROM (Read Only Memory) e ai file contenuti sui normali dischi magnetici flessibili (floppy). E' inoltre possibile aggiungere al computer sistemi di archiviazione opzionali per permettergli di funzionare, ad esempio, come workstation di una rete. In ogni caso, per l'utente principiante sarà più che sufficiente apprendere solamente il sistema ADFS.

All'accensione, o in seguito ad un hard break (CTRL + BREAK), il computer seleziona automaticamente il sistema di archiviazione designato che diventa il sistema di archiviazione corrente e rimane attivo finchè non si comunica al MOS, il sistema operativo, (utilizzando i comandi descritti di seguito) che si desidera utilizzare un altro sistema. Il sistema di archiviazione standard, designato alla prima accensione del computer, è lo ADFS.

Vale la pena ricordare che ogni sistema di archiviazione è, nei limiti del possibile, compatibile con gli altri sistemi. Questo significa che il comando per caricare un file in memoria funziona correttamente sia su RFS, sia su ADFS, sia sul sistema di archiviazione di rete. Questa compatibilità facilita l'uso del computer e permette di ridurre al minimo i comandi da utilizzare per il funzionamento del computer. Naturalmente alcuni sistemi di archiviazione possiedono funzioni o comandi che non possono essere utilizzati da altri sistemi. Ad esempio non è possibile salvare (SAVE) un file in un sistema di archiviazione ROM a causa della natura del dispositivo di memorizzazione (*sola lettura*).

Il linguaggio BASIC ed altri programmi applicativi disponibili per questo computer possiedono propri comandi incorporati per la comunicazione con il sistema di archiviazione. Per dettagli consultare la documentazione fornita con i programmi. Di seguito viene fornita una descrizione del modo in cui i file vengono memorizzati da ciascun sistema di archiviazione e una spiegazione dei comandi necessari per l'utilizzo del sistema di archiviazione ad un livello elementare. Per la lista completa dei comandi vedere l'Appendice 7.

Sistema di archiviazione ROM (RFS)

Selezionato con: *ROM RETURN

Il sistema di archiviazione ROM (memoria a sola lettura) permette di accedere ai file contenuti nei chip ROM, che vengono inseriti all'interno del computer. Memorizzazione dei file nei chip ROM significa che questi file sono a disposizione del computer e le informazioni in essi contenute possono essere lette immediatamente dal computer.

I chip ROM sono essenzialmente una forma permanente di memorizzazione e, data la mancanza in questi chip di parti mobili, sono semplici e molto affidabili. Per questi motivi vengono utilizzati per memorizzare le informazioni separatamente dal sistema di archiviazione. Ad esempio, i linguaggi BASIC e il sistema operativo MOS sono memorizzati in memorie a sola lettura all'interno del computer.

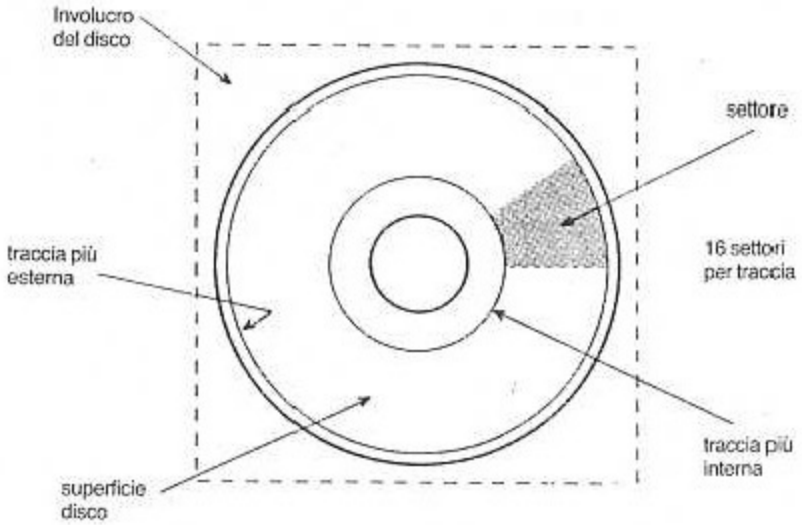
Naturalmente, nonostante il suo nome, è possibile scrivere sulla memoria di sola lettura, in genere in fase di fabbricazione e con un programmer speciale.

Sistema di archiviazione avanzato su disco (ADFS)

Selezionato con: *ADFS RETURN

Dischi

Il sistema di archiviazione avanzato su disco (ADFS) è il metodo di memorizzazione esterna fornito dal computer. Il termine "avanzato" si riferisce alle sue caratteristiche e prestazioni piuttosto che alla difficoltà del suo utilizzo. ADFS fornisce una potente funzione di memorizzazione file su dischetti da 3,5". I dischetti sono formati da una pellicola plastica rivestita di materiale magnetico contenuta in un involucro rigido di protezione. I file vengono memorizzati sul disco in cerchi concentrici chiamati **tracce**, ciascuna delle quali è suddivisa in 16 **settori**:



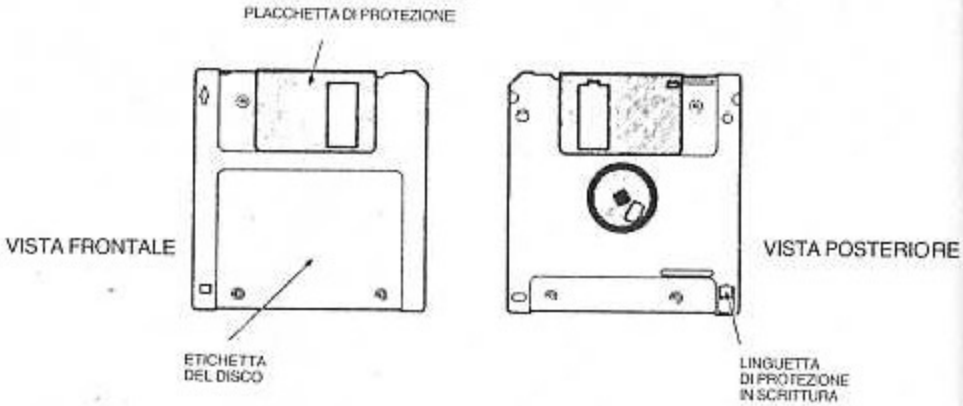
I dischi utilizzati con questo computer hanno 80 tracce e sono a **doppia faccia** (vale a dire che su entrambe le facce si possono memorizzare i dati). Il computer legge il dischetto per mezzo di un'unità a disco o **disk drive**. Questo drive fa ruotare il disco ad una velocità prefissata e appoggia sullo stesso le **testine di lettura/ scrittura** che, per poter memorizzare le informazioni, alterano le proprietà magnetiche di parti del disco. La capacità del disco completo è di 640 Kbyte.

Il computer viene venduto con un disk drive, montato sul lato sinistro della macchina e che viene denominato drive 0. E' comunque possibile acquistare un secondo drive che verrà montato a destra. Questo secondo drive viene denominato drive 1 ed è molto utile quando si ha la necessità di lavorare con grossi quantitativi di dati.

Il computer è inoltre fornito di un disco, il disco Welcome, che contiene molti programmi di esempio e di utilità. Osservando il disco si potranno notare:

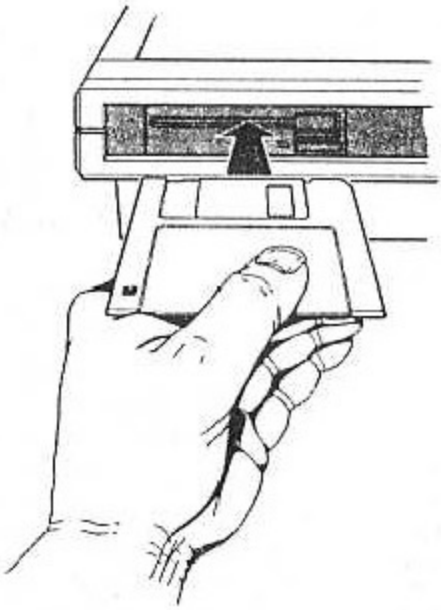
- Una placchetta in metallo su un lato del disco che lo protegge dalla polvere quando non viene utilizzato. Facendo scorrere questa placchetta di lato si potrà vedere il disco vero e proprio. *Fare attenzione a non toccare il disco.*

- Sul lato opposto è presente una linguetta in plastica chiamata **linguetta di protezione in scrittura** che può essere fatta scorrere di lato. Questa linguetta permette di evitare che il computer scriva qualsiasi informazione sul disco ad esempio quando si desidera mantenere intatto quanto contenuto sul disco senza correre il rischio di cancellature accidentali. Questa linguetta è simile a quella presente sulle cassette magnetiche, che ha lo stesso scopo.



Il disco non può essere scritto o cancellato quando la linguetta è posizionata in basso (verso l'esterno), cioè quando la fessura creata dalla linguetta è aperta.

La figura riportata di seguito mostra come inserire correttamente il disco nel disk drive:



Gestione dei dischi

Prima di passare ad informazioni dettagliate su ADFS, è opportuno ricordare alcune precauzioni. I dischetti sono strumenti molto delicati e devono essere trattati con la massima cura ed attenzione. I dischetti possono essere danneggiati in due modi: fisicamente e magneticamente.

- Il *danno fisico* è provocato piegando l'involucro del dischetto, toccando la superficie del disco oppure dal contatto di polvere o liquidi con il disco. Riporre sempre i dischetti su piani più alti rispetto al tavolo di lavoro per evitare che entrino in contatto con bevande e in una scatola a prova di polvere, per evitare il contatto con polvere e fumo. Non esporre i dischetti a sbalzi di temperatura e alla luce diretta del sole. Anche il disk drive è sensibile a urti, quindi deve essere tenuto su un piano orizzontale non inclinato in modo che non scivoli.
- Il *danno magnetico* non si provoca facilmente ma è ugualmente pericoloso. Non lasciare i dischi vicino ad apparecchiature elettriche ed in particolar modo evitare di avvicinarlo a fonti magnetiche quali calamite, trasformatori, sistemi hi–fi e televisori. Non estrarre mai un disco dal drive quando la spia del drive stesso è accesa per non rischiare di perdere le informazioni sul disco.

Seguendo queste precauzioni si verrà ripagati da una lunga durata e da una buona affidabilità dei dischetti riducendo al minimo la possibilità di perdite di dati preziosi.

Formattazione di un disco

Quando si acquista un dischetto, questo è completamente vuoto, cioè non è formattato. I dischi vergini non possono essere utilizzati per la memorizzazione dei dati finché non vengono **formattati**. La formattazione crea una struttura di informazioni sul disco permettendo così al computer di trovare i dati sulla superficie del dischetto. Notare che l'operazione di formattazione cancella completamente tutti i dati contenuti nel disco, quindi, se si formatta un disco non vuoto, i dati in esso contenuti saranno persi per sempre e non potranno essere recuperati in nessun modo.

Il disco Welcome è già formattato e pieno di dati e può essere utilizzato subito.

È consigliabile tenere il disco Welcome il più possibile protetto in scrittura in modo da evitare il rischio di cancellature accidentali.

Quando si acquista un dischetto è necessario formattarlo con il comando *FORMAT prima di utilizzarlo. Si dovrà comunicare al computer il drive in cui si trova il disco da formattare e la quantità di dati che può contenere. I dischetti possono essere di tre dimensioni: Large, Medium e Small ma, a meno che non vi sia una necessità particolare, si consiglia di utilizzare sempre la dimensione più grande (L) che utilizza le due facce del disco. Per formattare un dischetto rendendolo Large nel drive a sinistra (drive 0), digitare quanto segue:

*FORMAT 0 L **RETURN**

Prima dell'avvio dell'operazione di formattazione, sullo schermo appare il messaggio:

Format drive :0 ?

Introdurre YES per avviare la formattazione. La pressione di un tasti diversi da YES provoca l'annullamento dell'operazione.

Se la linguetta è in posizione di protezione, sullo schermo viene visualizzato il messaggio di errore:

Disc protected
(Disco protetto)

Spostare la linguetta di protezione e ripetere il comando di formattazione.

Al termine della formattazione, ADFS **verificherà** che l'operazione si sia svolta con successo. Con questo processo il computer controlla ogni traccia e visualizza un simbolo "?" se si è verificato un problema. I dischi su cui si verificano errori software indicati da uno o più punti interrogativi devono essere riformattati. Se però il disco è veramente difettoso, il programma visualizzerà:

Disc error 08 at 000201
(errore disco 08 a 000201)

ed il dischetto dovrà essere gettato. Il disco può essere verificato in qualsiasi momento per mezzo del comando *VERIFY, ad esempio se si sospetta che un disco sia difettoso.

Copie di sicurezza di un disco

Sebbene i dischi siano abbastanza affidabili, si possono verificare circostanze esterne come ad esempio cadute di tensione che possono causare la perdita di informazioni. Per salvaguardarsi da questi pericoli si consiglia di eseguire sempre una copia di riserva (o di **backup**) dei dischi che contengono dati importanti. Il costo dovuto alla perdita dei dati è di gran lunga superiore al costo di un dischetto. Per eseguire copie di riserva, ADFS dispone del comando *BACKUP. Se non lo si fosse ancora fatto, eseguire al più presto una copia del disco Welcome. La sintassi del comando *BACKUP è la seguente:

*BACKUP 0 0 RETURN

I due numeri che seguono il comando vero e proprio rappresentano rispettivamente il numero del drive sorgente (source) e quello del drive di destinazione (destination). Se si possiede un solo drive i due numeri saranno uguali, come mostrato nell'esempio e ADFS richiederà di scambiare i dischi nel corso del processo di copiatura. Naturalmente il disco di destinazione sarà un disco appena formattato, per evitare di perdere dati preziosi. Si consiglia di apporre la protezione in scrittura sul disco che verrà copiato per evitare che venga alterato. Il computer chiede di confermare che si desidera sovrascrivere il disco di destinazione visualizzando il messaggio:

Backup drive :0 to :0 ?

Rispondere con YES.

Se si è in possesso di due disk drive sarà possibile copiare un disco su un altro senza effettuare lo scambio dei dischi, ma introducendo il disco originale in un drive ed un disco appena formattato nell'altro drive. A questo punto digitare:

*BACKUP 0 1 RETURN

Se desiderate fare copie di backup con un solo drive. Usate il comando *SDBACKUP. Esso copia dischi in formato ADFS utilizzando 80 Kbytes di memoria del computer. Questo programma di utilità può essere eseguito sia a partire dal menu Welcome selezionando Utilità e ADFS, sia direttamente dalla libreria del vostro disco Welcome, digitando:

*SDBACKUP RETURN

Prima dell'operazione accertarsi che il disco DESTINATION non sia protetto; non verrebbe copiato alcunchè dal disco SOURCE.

Quando la copia è terminata premete il tasto BREAK.

Se volete ricaricare il programma Welcome inserite nel drive il disco Welcome e premete:

SHIFT + BREAK

Nota: si raccomanda di fare una copia di backup del disco Welcome il più presto possibile.

Uso di ADFS

Prima di proseguire con la lettura di questa sezione vorremmo ricordare che sul disco Welcome è presente un programma che spiega come usare ADFS. Per caricare questo programma selezionarlo dal menu principale del programma Welcome, illustrato all'inizio di questo manuale.

Inizializzazione di ADFS

ADFS è il sistema di archiviazione selezionato automaticamente all'accensione del computer. A questo punto occorre inserire il disco Welcome nel drive, con l'etichetta rivolta verso l'alto, facendo entrare per prima la placchetta in metallo. E' impossibile inserire un dischetto in modo errato. Non forzare MAI un disco nel drive, altrimenti si potrebbero provocare danni al disco e al drive stesso. Dopo aver introdotto il disco, ADFS inizierà a leggere i dati in esso contenuti. L'operazione di lettura dura alcuni secondi.

Per utilizzare un secondo disk drive sarà necessario avviare ADFS senza accedere al drive e quindi comunicare al computer con il comando *MOUNT il drive in cui si trova il dischetto. Digitare:

*FADFS RETURN
*MOUNT 1 RETURN

Il comando *FADFS prepara ADFS ma non provoca l'accesso al drive. Il comando *MOUNT 1 indica a ADFS di lavorare con il dischetto nel drive 1. Per tornare ancora sull'altro drive, digitare:

*DISMOUNT RETURN

Per abbandonare il drive 1 definitivamente, battere ancora il comando che segue per indicare a ADFS il drive in cui si trova il disco:

*MOUNT 0 RETURN

Prima di estrarre un disco dal drive utilizzare il comando *DISMOUNT che verifica che tutti i dati siano stati aggiornati. Non estrarre mai dal drive un disco quando la spia è accesa: si potrebbero causare danni al disco e al drive stesso.

Quando si utilizza un programma o un programma di utilità sul disco Welcome assicurarsi che il disco Welcome sia installato e pronto per l'uso. Per eseguire questa operazione, digitare i seguenti comandi prima di tentare di caricare o salvare dei dati:

*MOUNT 0 RETURN
*LIB LIBRARY RETURN

Il significato del secondo comando verrà spiegato più avanti in questo capitolo.

Salvataggio di un file con ADFS

Il concetto di file esiste per facilitare il riferimento ad informazioni memorizzate su dischi che ad occhio nudo sono invisibili. Un file è semplicemente una sequenza di byte (caratteri) memorizzati su disco invece che nella memoria del computer. Un file può avere una lunghezza minima di 0 byte (file vuoto) ed una lunghezza massima di poco più di 600 Kbyte, limite imposto dalla capacità del disco. Ad ogni file viene attribuito un nome, che può avere una lunghezza massima di 10 caratteri. I **nomi di file** possono contenere lettere maiuscole e minuscole e numeri (sebbene ADFS tratti tutte le lettere come se fossero maiuscole quando fa riferimento ad un file). Per il nome possono essere utilizzati anche simboli di punteggiatura, ma per il momento si consiglia di non usarli in quanto alcuni di questi simboli possono avere significati speciali per ADFS.

Per visualizzare i nomi dei file contenuti nel disco, ADFS dispone del comando di catalogo *CAT, che può essere abbreviato semplicemente in *, in quanto sarà il comando digitato più frequentemente. Assicurarsi che il disco Welcome sia presente nel drive e che sia pronto per l'uso, quindi battere:

*CAT RETURN

Sullo schermo dovrebbe apparire una tabella simile quella riportata di seguito:

Welcome xxxxx	(xx)
Drive:0	Option 03 (Exec)
Dir. $	Lib. "Library"

```
!BOOT      LWR(10)    LIBRARY    DLR(12)
WELCOME    DLR(11)    UTILS      DLR(06)
```

Le ultime righe mostrano i nomi dei file memorizzati sul disco. Al di sopra di queste due righe sono presenti alcune informazioni secondarie tra cui il nome del disco.

Proviamo ora a salvare (comando SAVE) un programma BASIC. Si prega di utilizzare a questo scopo un nuovo disco e non il disco Welcome, per non cancellare accidentalmente i programmi memorizzati su quest'ultimo. Digitare quanto segue, inserendo un RETURN alla fine di ogni riga:

```
NEW
10 FOR I%=32 TO 126
20 VDU I%
30 NEXT I%
40 PRINT
```

Salvare ora il programma nel solito modo:

SAVE"PROVA" RETURN

La spia del drive rimarrà accesa per alcuni istanti, quindi ricomparirà il messaggio BASIC ('>'). Per verificare che il programma sia stato salvato correttamente, digitare un altro comando "CAT. Questa volta comparirà il nome "PROVA" nella lista dei file (i nomi sono ordinati alfabeticamente) seguito da 'WR'. Queste lettere indicano le operazioni permesse su questo file: 'W' sta per scrittura e 'R' sta per lettura (il file può essere scritto e caricato in memoria). Più avanti vedremo lo scopo di queste due funzioni e come modificarle.

Per verificare ulteriormente che il file si trovi realmente sul disco, battere NEW RETURN per cancellare il programma dalla memoria seguito da:

LOAD "PROVA" RETURN

La spia del drive si illuminerà ancora e sullo schermo riapparirà il messaggio BASIC ('>'). Digitando a questo punto LIST RETURN si noterà che il programma è stato copiato dal disco alla memoria.

Ricordare che salvando un file utilizzando il nome di un file già presente sul disco, il nuovo file sovrascriverà sempre il file precedente. Per comprendere meglio questa funzione, aggiungere al programma 'PROVA' la riga seguente:

1 REM Questa è una nuova versione di 'PROVA' RETURN

e salvare il file:

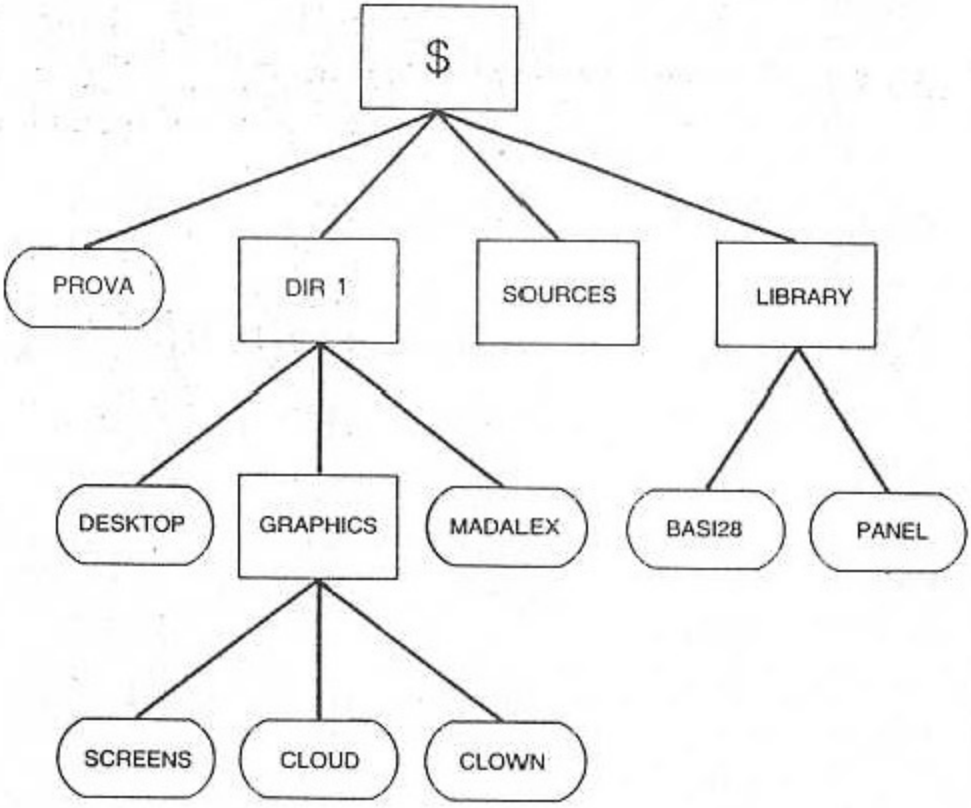
SAVE"PROVA" RETURN

La versione precedente del programma viene così sostituita dalla nuova versione e potrà essere controllata eseguendo i passi precedenti. Questo dimostra che se si riutilizza un nome già esistente, il vecchio file viene *sempre* sovrascritto. Da qui la necessità di scegliere nomi diversi per file sullo stesso disco.

Directory

Si può facilmente immaginare che possono capitare circostanze in cui potrebbe essere utile riutilizzare dei nomi di file. Dopo aver salvato diversi file diventa problematico verificare se un nome è già stato utilizzato per un altro file. Per questo ed altri motivi ADFS fornisce le **directory** che permettono di aggirare questo ostacolo.

E' più semplice comprendere le directory aiutandosi con un'analogia. Immaginiamo un file come se fosse un foglio di carta, con il nome scritto in alto. Per archiviare i fogli di carta si sceglieranno i fogli che riguardano lo stesso argomento e li si raggrupperà in una cartellina assegnando alla stessa un nome. In questo modo si sarà facilitati nella ricerca di un foglio di carta e si potranno ordinare ulteriormente i fogli inserendo cartelline una all'interno dell'altra. ADFS permette di creare file speciali, chiamati directory che contengono altri file, esattamente come avviene con le cartelline. Anche le directory possono contenere altre directory, chiamate **sub–directory**, che a loro volta contengono file e sub–directory. Questa collezione di oggetti nidificati uno all'interno dell'altro viene chiamata **gerarchia** o albero e può essere compresa più facilmente osservando la figura qui sotto:



In alto nella gerarchia è presente una directory chiamata **radice** (abbiamo disegnato l'albero capovolto per facilitare la comprensione) che contiene tutti i file e le directory presenti sul disco. La radice è rappresentata in ADFS con il simbolo $. In questo esempio la radice contiene diverse sub–directory, tra cui LIBRARY e DIR1 che a loro volta contengono dei file. Per visualizzare il contenuto di $ è possibile utilizzare il comando *CAT.

All'avviamento di ADFS e quando si impartisce il comando *MOUNT, la directory $ diventa automaticamente la "directory selezionata correntemente" o CSD. Digitando *CAT, ADFS produce un catalogo della CSD. Per visualizzare invece cataloghi di altre directory, far seguire il comando *CAT dal nome di una directory, cioè ad esempio:

*CAT LIBRARY oppure
*CAT DIR1

Utilizzando *CAT in questo modo si visualizza soltanto il contenuto di un'altra directory ma, per caricare un file da una sub–directory occorre essere più precisi in quanto ADFS assume che si stia cercando un file nella directory selezionata correntamente.

Precedentemente avevamo salvato il file "PROVA" nella directory '$'. Per specificare dove ricercare questo file occorre introdurre il suo nome completo (o **nome di percorso**). Nel caso di PROVA il nome completo è '$.PROVA'. Il punto serve a separare il nome del file dal nome di directory e se necessario possono essere utilizzati più punti. Quelli che seguono sono esempi di nomi in base alla figura precedente:

$.DIR1	Sub–directory 'DIR1' nella directory '$'
$.LIBRARY.PANEL	File 'PANEL' nella directory $.LIBRARY

Si sottintende che il nome della directory selezionata correntemente precede qualsiasi nome di file che non dipenda direttamente da '$', ed è possibile includerlo in un nome di percorso utilizzando il simbolo '@'.

Uso della gerarchia di directory

Perché le directory possano essere utili, è necessario avere la possibilità di modificare la directory corrente e creare nuove directory. Il comando *DIR viene utilizzato per modificare la CSD. Ad esempio i due comandi che seguono rendono la radice la directory corrente:

*DIR

*DIR $

Si possono utilizzare nomi di percorso di qualsiasi lunghezza, come ad esempio:

*DIR $.LIBRARY.BAS128

Per tornare alla directory che contiene la CSD, inserire il simbolo di accento circonflesso '^', la cui somiglianza ad una freccia rivolta verso l'alto non è puramente una coincidenza. Supponiamo che la CSD sia molto in basso in una gerarchia, che abbia ad esempio il nome di percorso '$.Lettere.Commerciali.Maggio.AllaXYZ'. Per spostarsi in alto di un livello (in base al diagramma ad albero) si dovrà digitare:

*DIR $.Lettere.Commerciali.Maggio

Se si utilizza l'accento circonflesso, la riga precedente potrà essere abbreviata in:

*DIR ^

Un effetto simile si può ottenere con il comando *BACK, che riporta alla directory selezionata precedentemente, indipendentemente dal punto in cui si trovava nella gerarchia. Questo comando non ha sempre l'effetto di *DIR ^, sebbene si possa cercare di immaginare la sequenza di eventi che lo rendono tale.

Nel caso si utilizzassero due drive, i nomi di percorso accettati da *DIR possono includere il numero di drive preceduto da un simbolo di due punti (':'). Ad esempio, per catalogare la directory $.GAMES sul disco nel drive 1 occorrerà digitare:

*CAT :1.$.GAMES oppure

*CAT :1.GAMES (il simbolo $ è sottinteso)

Questo significa che non è necessario spostare i dischi da un drive all'altro per visualizzare o caricare i file.

Creazione di directory

E' possibile creare nuove directory in qualsiasi punto della gerarchia. Per questa operazione si utilizza il comando *CDIR, seguito dal nome di percorso della nuova directory:

*CDIR $.Lettere.AMarco

Questo creerà una directory chiamata 'AMarco' in '$.Lettere'. Se '$.Lettere' fosse la directory selezionata correntemente, si potrebbe digitare:

*CDIR AMarco

Dopo aver creato una directory si potrà renderla corrente per mezzo del comando *DIR. Catalogando una nuova directory con *CAT, questa non conterrà alcun file finché al suo interno non ne verranno memorizzati esplicitamente.

Librerie

Vi sono tre comandi per lanciare programmi in linguaggio macchina: (Assembler)

*RUN nome_del_file
*/ nome_del_file
*nome_del_file

Il primo è la forma corretta, il secondo è un'abbreviazione. Il comando *RUN provoca il caricamento del file in memoria e lo esegue. Il file specificato viene ricercato nella CSD e quindi in una directory speciale chiamata **libreria** (libreria selezionata correntemente **(CSL)**).

Il terzo formato del comando permette al MOS di intercettare la richiesta ed eseguire una versione ROM del programma, se esistente. In caso contrario la ricerca viene continuata nella CSD e nella CSL nel modo normale. La libreria serve, in pratica, ad estendere sul disco i comandi del sistema operativo basati su ROM.

Per specificare la directory **libreria**, si imposta la libreria selezionata correntemente con il comando *LIB:

*LIB $.Library **RETURN**

essendo $.Library il nome del CSL. Per convertire la CSD in CSL, utilizzare l'abbreviazione '@', anzichè scrivere per intero il nome della CSD e cioè:

*LIB @ **RETURN**

Attivando ADFS con *FADFS, la CSD e la CSL vengono entrambe disattivate. Introducendo *MOUNT la CSD viene impostata su '$', ma la CSL rimane disattivata e deve essere impostata con *LIB.

Il comando *LCAT permette di catalogare velocemente la CSL ed ha lo stesso effetto dell'introduzione di *CAT seguito dal nome di percorso completo della CSL.

Ulteriori informazioni sulle directory

Esamineremo ora in dettaglio le informazioni stampate con il comando *CAT. Supponiamo che venga visualizzato quanto segue:

```
Main disc                (69)
Drive:0                  Option 00 (Off)
Dir.$                    Lib. $.Library

BOUNCE     LR(01)        Capricorn   DLR(03)
Games     DLR(06)        Library     DLR(12)
PROVA      WR(13)        XX           EL(05)
```

Le prime tre righe contengono informazioni generali sulla directory e ADFS, mentre le righe successive forniscono informazioni sui file individuali all'interno della directory.

La prima riga contiene il titolo ed il numero di sequenza principale (MSN) del disco. Il titolo del disco è una stringa composta da un massimo di 19 caratteri che generalmente descrive il contenuto del disco. Il titolo della CSD viene impostato con il comando *TITLE. Il nome viene comunque impostato per default a seconda del nome della directory. Nell'esempio il titolo era Main disc (disco principale).

Il numero di sequenza principale è zero quando il disco è formattato. Quando un file viene salvato nella directory, il numero di sequenza principale viene aumentato di uno e questo numero viene memorizzato nell'introduzione di numero di sequenza del file (vedere più avanti). Il numero permette di conoscere la sequenza in cui sono stati memorizzati i file. I numeri vengono riportati a 00 quando si raggiunge il valore 99.

La seconda riga del listato fornisce informazioni sul drive corrente. Il numero del drive è generalmente 0 o 1. "Option" indica a ADFS cosa fare con il file !BOOT dopo che è stata utilizzata la sequenza di tasti SHIFT + BREAK. L'impostazione di default è 'Off' (disattivato).

Una caratteristica importante del PC 128 S è il file !BOOT. Questo contiene una serie di istruzioni che predispongono il computer a eseguire una determinata sequenza di operazioni iniziali.
Queste istruzioni, accessibili e modificabili dall'utente, vengono eseguite ogni qualvolta sono premuti i tasti SHIFT + BREAK.

La terza riga del listato fornisce i nomi della directory e della libreria selezionate correntemente.

La parte rimanente del listato fornisce i nomi dei file contenuti nella directory, il loro 'codice di accesso' ed i loro numeri di sequenza individuali. I codici di accesso verranno descritti dettagliatamente nella sezione riguardante il comando *ACCESS qui sotto.

Informazioni dettagliate di un file

ADFS memorizza più informazioni riguardo ad un file di quante ne vengono visualizzate quando si cataloga una directory. Il comando *INFO visualizza queste informazioni supplementari se lo si fa seguire dal nome di un file. Il comando *EX invece stampa le informazioni di tutti i file nella CSD o in una sub–directory specificata.

Il risultato di un comando *INFO potrebbe apparire come segue:

Computer	DLR(03)	0001FE			
SCARTI	LWR(09)	00000000	FFFFFFFF	00000EFE	00009C
Lettere	DLR(07)	00004A			

La prima parte della riga è uguale a quella di un catalogo, mentre l'ultima parte è diversa a seconda che l'elemento specificato sia un file o una directory.

I file sono seguiti da quattro elementi di informazione, ciascuno espresso in notazione esadecimale. I quattro elementi sono: l'indirizzo di caricamento, l'indirizzo di esecuzione, la lunghezza del file e l'indirizzo di disco del file. L'indirizzo di caricamento indica a ADFS dove caricare il file nella memoria, l'indirizzo di esecuzione indica dove iniziare l'esecuzione del programma dopo il caricamento e la lunghezza è semplicemente il numero di byte occupati dal file. Tutti questi valori possono essere stampati in notazione decimale utilizzando il BASIC, ad esempio:

PRINT &EFE RETURN

visualizzerà la lunghezza del file SCARTI in notazione decimale.

L'indirizzo del disco viene utilizzato internamente da ADFS per ricercare il file sulla superficie del disco e generalmente non viene utilizzato dagli utenti.

Le directory visualizzano esclusivamente i propri indirizzi del disco; gli altri dati non sono significativi.

Esiste anche un formato speciale del comando *EX: *LEX, che produce informazioni dettagliate sui file nella CSL.

Wildcard

ADFS supporta una funzione per l'uso di speciali caratteri detti **wildcard** nei nomi di file. Questi caratteri possono essere utilizzati per specificare nomi di file ambigui, vale a dire nomi di file che possono essere scritti in diversi modi permettendo così di svolgere diverse operazioni su diversi file utilizzando un solo comando.
Il carattere wildcard singolo '#' indica un carattere non specificato, mentre il carattere wildcard multiplo è '*'. Di seguito vengono mostrati alcuni esempi di nomi di file contenenti wildcard:

Capitolo#	(ad es. 'Capitolo1', 'CapitoloG')
GIORNO##	(ad es. 'GIORNOaa', 'GIORNO36')
$.LETTERE.*	(qualsiasi file in '$.LETTERE')
FISICA.*Calorie*	(qualsiasi nome di file in 'FISICA' contenente 'Calorie')

Il modo in cui i nomi di file contenenti caratteri wildcard vengono interpretati dipende dal comando utilizzato. La maggior parte dei comandi agirà sul primo file che soddisfa il modello specificato di wildcard in ordine alfabetico. Il comando *CAT è uno di questi. Esempio:

CAT Q

catalogherà la prima directory nella CSD che inizia con una 'Q' o una 'q' (ricordare che ADFS non fa distinzione tra lettere maiuscole e minuscole).

Altri comandi agiranno su tutti i nomi di file che corrispondono al modello di wildcard, come ad esempio *INFO. Quindi:

INFO Capitolo

visualizzerà le informazioni riguardo a tutti i file il cui nome inizi con 'Capitolo'.

Un terzo gruppo di comandi non ammette wildcard e visualizza il messaggio di errore 'Wildcards' se si tenta di utilizzarli. Uno di questi comandi è *DELETE, che verrà illustrato più avanti.

Codici di accesso ed il comando *ACCESS

*CAT, *INFO ed i loro comandi associati stampano delle lettere tra il nome del file ed il numero di sequenza, detti codici di accesso, che descrivono il tipo di operazioni eseguibili sul file. Le lettere possibili sono cinque, ciascuna delle quali deve essere impostata (e quindi stampata da *CAT/*INFO) oppure disattivata (e non stampata):

D – Directory. La presenza di questa lettera indica che il file è una directory. Viene impostata alla creazione del file e non può più essere modificata. I file directory non possono essere caricati o eseguiti.

E – Execute (esecuzione). Se è presente questo codice, il file (non directory) non può essere caricato o letto in alcun modo, ma può solamente essere eseguito.
I comandi che funzionano su questo tipo di file sono *RUN, */, *ACCESS, *DELETE, *REMOVE e *DESTROY. Questo codice, se impostato, non può più essere cancellato. L'unico codice che può sostituire 'E' è 'L' (vedere di seguito). *INFO produce informazioni essenziali su questi file.

L – Locked (bloccato). I file bloccati non possono essere cancellati da *DELETE, *REMOVE o *DESTROY e non possono essere ridenominati con *RENAME o sovrascritti in alcun modo (vedere più avanti). Qualsiasi tentativo di cancellazione senza prima rimuovere il codice 'L' per mezzo di *ACCESS provocherà il messaggio di errore 'Locked'. Per le directory viene sempre utilizzato per default il codice 'L' in fase di creazione, mentre sui file l'utente può impostare o cancellare questo codice.

R – Read (lettura). Le informazioni possono essere lette dal file solamente se è presente questo codice. E' difficile che un utente decida di cancellare questo codice in quanto normalmente tutti i file devono essere letti. Il carattere di accesso 'R' non può essere cancellato da una directory.

W – Write (scrittura). Questo carattere determina se i dati possono essere scritti sul file. Se questo codice non è presente, i file non potranno essere nè scritti nè modificati in alcun modo (notare comunque che deve essere impostato il carattere 'L' per evitare cancellazioni accidentali del file). Le directory non possono avere il carattere 'W'.

Il comando *ACCESS permette di modificare i caratteri di codice di accesso del file. *ACCESS deve essere seguito da un nome di percorso (sono ammessi wildcard), seguito a sua volta dai codici di accesso da applicare ai file corrispondenti. Esempi:

*ACCESS prova	Annulla ogni codice di accesso a 'prova' ad eccezione di 'D' ed 'E'.
*ACCESS * LR	Protegge tutti i file della CSD da sovrascrittura e cancellazione.
ACCESS Lettere. WR	Rende possibile la lettura e la scrittura per i file in 'lettere'.

Le applicazioni più comuni dei codici di accesso sono l'impostazione di 'L' per prevenire una cancellazione accidentale, la cancellazione di 'L' per permettere una cancellazione voluta e l'impostazione di 'W' per evitare una sovrascrittura accidentale.

Cancellazione, ridenominazione e copia di un file

Quando un file non è più necessario, dovrebbe essere cancellato, questo per liberare lo spazio da esso occupato sul disco. Il comando *DELETE cancella un file da una directory. Il nome del file non deve contenere wildcard. Ad esempio:

*DELETE scarti

*DELETE $.!BOOT

Se il file specificato non esiste, sullo schermo appare il messaggio 'Not found' (non trovato). A volte, in special modo se si opera all'interno di programmi, sarebbe comodo poter verificare che un file non esiste senza però provocare un messaggio di errore se il file è già stato cancellato. Allo scopo viene utilizzato il comando *REMOVE che funziona esattamente come *DELETE, senza però visualizzare alcun messaggio se il file non esiste.

*DELETE e *REMOVE non hanno effetto su un file protetto con il codice di accesso 'L'. Per cancellare un file bloccato o protetto occorre per prima cosa rimuovere il codice 'L' per mezzo di *ACCESS. Ad esempio:

*ACCESS dati
*DELETE dati

Le directory sono sempre protette a meno che non si modifichi il loro codice di accesso, ma pongono un altro limite alla cancellazione: non devono contenere file. Il tentativo di cancellare una directory che contiene dei file provoca il messaggio di errore 'Dir. not empty' (directory non vuota).

Per cancellare un gruppo di file, ad esempio tutti i file di una directory, si utilizza il comando *DESTROY, che accetta nomi di file contenenti caratteri wildcard. Vengono cancellati tutti i file che rispondono alla specifica del carattere wildcard. Prima della cancellazione dei file il programma visualizza un'informazione *INFO per tutti i file interessati e quindi al disotto:

Destroy ?
(cancello?)

Per confermare premere YES, in caso contrario l'operazione viene annullata.

La possibilità di ridenominare i file è molto utile non soltanto perchè con questa funzione è possibile cambiare il nome di un file, ma soprattutto perchè, assegnando un altro nome ad un file viene modificata la gerarchia dei file. Il comando *RENAME deve essere seguito dal nome di percorso sorgente e dalla destinazione. Quindi il comando:

*RENAME questo quello

modificherà il nome del file chiamato 'questo' in 'quello'. Inoltre un comando quale:

*RENAME prova $.programmi.prova

sposterà il file 'prova' nella CSD nella directory '$.programmi', mantenendo il nome 'prova'. E' inoltre possibile modificare il nome del file mentre lo si trasferisce utilizzando un comando quale '^.':

*RENAME VecchioFile ^.NuovoFile

In questo modo il file 'VecchioFile' verrà spostato dalla directory selezionata correntemente alla directory radice e verrà chiamato 'NuovoFile'. Non utilizzare mai wildcard in un comando *RENAME.

Il comando *COPY esegue una copia di uno o più file, mantenendo la versione originale nel punto in cui si trovava. Il comando deve essere seguito dal nome del file sorgente (che può contenere wildcard) e dal nome di percorso del file destinazione (che non può contenere wildcard). *COPY non può copiare directory (in quanto a loro volta possono contenere altre directory, ecc.). Un esempio di copiatura potrebbe essere:

COPY $. $.Backup

che copia tutti i file non directory in '$' nella directory '$.Backup' mantenendo la versione originale.

*COPY utilizza tutta la memoria shadow che ha a disposizione e in genere la preferisce alla memoria principale.

Se si è in possesso di due disk drive, si potranno copiare i file da un drive all'altro includendo nel nome di percorso il numero di drive, ad esempio:

COPY :0. :1

che copia tutti i file non directory dalla radice nel drive 0 alla radice nel drive 1. Questa operazione *non* è possibile con un solo drive in quanto *COPY non emette richieste di scambio dischi. Per poter copiare da un disco all'altro utilizzando un solo drive, oppure se si desidera copiare directory intere (comprese tutte le sub–directory che queste possono contenere) si dovrà utilizzare il programma di utilità 'DIRCOPY' che si trova nel directory ADFS UTILS sul disco Welcome.

Comandi di controllo disco

I dischi hanno una capacità fissa, sebbene piuttosto ampia, di memoria. ADFS mette a disposizione dell'utente diversi comandi che permettono di sfruttare al massimo la capacità di un disco. Il primo comando che esamineremo è *FREE che indica esattamente quanto spazio disco è stato utilizzato e quanto è ancora disponibile. Digitando:

*FREE

ADFS visualizzerà una tabella simile alla seguente:

```
0006AA Sectors =    436,736 Bytes Free
000356 Sectors =    218,624 Bytes Used
```

Le prime cifre sono espresse in esadecimale e rappresentano il numero di settori da 256 byte ancora liberi. I file vengono allocati in multipli di settori pieni a seconda della loro dimensione, mentre le directory occupano cinque settori.

Il comando *MAP produce una lista piuttosto enigmatica delle aree del disco ancora libere. Introducendo un comando *MAP si potrebbe ottenere una tabella come la seguente:

```
Address :  Length
000002  :  00000A
00014A  :  000043
0001E3  :  00013D
```

La lista visualizzata potrebbe essere più lunga o più corta di quella dell'esempio ed i numeri saranno senz'altro diversi. Ciascun valore consiste di un indirizzo disco e della quantità di spazio libero in quella locazione. Entrambi i numeri sono espressi in esadecimale.

ADFS aggiorna automaticamente questa lista ogni volta che file e directory vengono creati o cancellati. Lo spazio libero è suddiviso in diversi settori del disco e per questo motivo un file potrebbe essere salvato, anche se lo spazio non è contiguo sul disco (prerequisito per la struttura di un file). Può rendersi a volte necessario **compattare** il disco. Il computer sottolinea questa necessità visualizzando il messaggio 'Compaction required'. Il compattamento sposta i file sul disco in modo che le aree vuote vengano riempite, lasciando un'area il più grande possibile di spazio contiguo.

Come il comando *COPY, anche il comando *COMPACT utilizza più memoria shadow possibile. Digitando il comando:

*COMPACT

la spia del drive si accende e rimane accesa per qualche minuto. Quando la spia si spegne, un comando *MAP visualizza grossi blocchi di spazio disponibile sul disco. A questo punto si può ritornare al modo di schermo precedente per proseguire il lavoro.

Comandi del sistema operativo MOS riguardanti i sistemi di archiviazione

Di seguito viene fornita una spiegazione dei comandi generali forniti dal sistema operativo MOS per l'utilizzo con qualsiasi sistema di archiviazione. Nell'esempio che segue si presuppone di utilizzare ADFS. Questi comandi mettono a disposizione funzioni avanzate che verranno sicuramente utilizzate dagli utenti esperti. **Per il momento i principianti potranno saltare questa sezione.**

Nota:

Si avvisa il lettore che, a causa della natura di *sola lettura* del sistema di archiviazione ROM (RFS) una parte di quanto segue non è rilevante per il suo utilizzo. In modo specifico l'RFS non utilizza comandi che *creano* o *modificano* i file, quindi non supporta *SAVE, *SPOOL, *BUILD, *CREATE o *APPEND.

Comandi file generali e programmi di utilità

I file possono essere considerati come copie su disco della memoria del computer. I file possono contenere programmi BASIC, testi trattati con word processor, programmi in linguaggio macchina, copie dello schermo oppure dati arbitrariamente complessi. I linguaggi come il BASIC dispongono di comandi per il salvataggio ed il caricamento di programmi, ma quando occorre salvare o caricare parti della memoria del computer si utilizzano i comandi del sistema di archiviazione MOS.

Il comando *SAVE salva il contenuto della memoria del computer direttamente sul disco. Questo comando ha diversi formati:

```
*SAVE RAM 0 8000
*SAVE area 3969 +452
```

Il primo comando salva il contenuto della memoria principale del computer, che si trova tra &0000 e &7FFF. Gli indirizzi forniti ai comandi del sistema di archiviazione o visualizzati da essi sono sempre espressi in notazione esadecimale. Il primo numero dopo il nome è l'indirizzo del primo byte da salvare; il secondo numero è l'indirizzo del byte dopo l'ultimo da salvare. I computer PC 128 S utilizzano il formato 'byte dopo' (ad es. HIMEM in BASIC), che facilita il calcolo della lunghezza.

Il secondo comando salva l'area di memoria tra &3969 e &3DBB (&3969+&452) nel file chiamato 'area'. Questo formato del comando utilizza un simbolo '+' seguito dalla dimensione dell'area di memoria.

Visto che si è utilizzato un comando *SAVE, si potrebbe voler visualizzare le informazioni relative al file (comando *INFO). Provare a digitare i comandi seguenti in BASIC:

```
>*SAVE pezzo 1C00 +200 RETURN
>*INFO pezzo RETURN
pezzo    WR(15)  00001C00  00001C00  00000200  000FE2

Name     Access  Load      Execute   Length    Disc address
```

Le voci 'Load', 'Execute' e 'Length' sono molto interessanti. 'Load' è l'indirizzo di memoria a cui il file verrà ricaricato (per mezzo del comando *LOAD, come vedremo più avanti). Notare che questo indirizzo è uguale all'indirizzo iniziale in cui è stato salvato il blocco di memoria, quindi verrà introdotto nello stesso punto. La voce 'Execute' è l'indirizzo che verrà richiamato se il file viene eseguito con *RUN o con un comando equivalente. Per default 'Execute' è uguale all'indirizzo di caricamento, ma può essere modificato, se necessario. La voce 'Length' è la lunghezza del file in byte fornita con il comando. Utilizzando il formato alternativo, MOS la calcolerà automaticamente. L'ultima voce è l'indirizzo disco, che al momento non ci interessa.

A volte può essere utile che l'indirizzo di esecuzione del file sia diverso dal suo indirizzo di caricamento, magari perchè si tratta di un programma in linguaggio macchina preceduto da dati. Questo si ottiene aggiungendo in fondo al comando *SAVE l'indirizzo di esecuzione. Ad esempio:

```
*SAVE Energia 2000 +1000 2100
```

In questo caso l'indirizzo di esecuzione è &2100, che supera il valore di default (che è l'indirizzo di caricamento).

L'estensione del comando *SAVE serve per specificare un indirizzo di caricamento diverso, cioè l'indirizzo di ricaricamento per il file. Ad esempio:

*SAVE MiaROM 4000 +2000 8002 8000
*SAVE Utilità 1900 1980 910 900

Il primo salva la memoria da &4000 a &5FFF nel file 'MiaROM' e fornisce un indirizzo di ricaricamento di &8000 ed un indirizzo di esecuzione di &8002. Il secondo salva il programma 'Utilità' tra gli indirizzi &1900 e &197F con un indirizzo di caricamento di &900 ed un indirizzo di esecuzione di &910. Notare che è possibile specificare solamente un indirizzo di ricaricamento se è presente anche l'indirizzo di esecuzione.

Il comando *LOAD ha l'effetto inverso di *SAVE ed ha due varianti. Nella prima variante *LOAD è seguito semplicemente dal nome del file e le altre informazioni sono prelevate dalla directory. Nella seconda variante l'indirizzo di caricamento può essere superato:

*LOAD MiaROM
*LOAD Utilità 700

Come abbiamo visto precedentemente, l'esecuzione di un programma in linguaggio macchina può essere effettuata con i tre diversi formati del comando *RUN:

*RUN nome_del_file
*/nome_del_file
*nome_del_file

dove 'nomefile' è un nome di percorso che può contenere caratteri wildcard (interpretati utilizzando il primo file corrispondente). *RUN e */ sono equivalenti. Questi due formati ricercano il file specificato nella CSD e quindi nella CSL, lo caricano e lo eseguono. Se il file non viene trovato, sullo schermo appare il messaggio 'Bad command' (comando errato). Il terzo formato per prima cosa verifica se il nome corrisponde al comando ROM. Se il nome non corrisponde, viene generato il messaggio di errore 'Bad command'.

Se ad un file viene assegnato l'indirizzo di esecuzione &FFFFFFFF, non verrà caricato ed eseguito in seguito a questi comandi, ma verrà eseguito dal comando *EXEC, come vedremo più avanti.

Gestione di schermo e tastiera

PC 128 S dispone di diversi comandi che gestiscono il modo in cui il computer comunica con l'utente. Il primo di questi comandi, *SPOOL, dirige una copia dell'output che appare sullo schermo verso il file specificato che si trova sul disco. L'operazione si effettua tramite i seguenti comandi:

*SPOOL nome_del_file

E con questo comando si predispone il computer a copiare l'output di schermo che viene copiato sul file finché non viene eseguito il comando:

*SPOOL

(senza nome di file). Quanto sopra è utile per creare un record contenente la registrazione del prodotto di un programma, in particolar modo quando questo riempie più schermi.

Il comando *TYPE visualizza il contenuto di questo file, visualizzando la registrazione effettuata con *SPOOL. Ad esempio:

*TYPE screen

Un formato alternativo di questo comando è *LIST, che visualizza il file allo stesso modo, ma aggiunge numeri di riga (a partire dal numero 0001 in su) davanti ad ogni riga visualizzata.

Un comando che ha grosso modo l'effetto opposto di *SPOOL è *EXEC. Questo comando legge il contenuto di un file *esattamente come se fosse stato introdotto da tastiera*, disattivando la tastiera nel corso dell'operazione. ADFS conosce già la lunghezza del file (in base alle informazioni della directory), quindi non sarà necessario arrestare l'esecuzione che, se lo si desidera, può comunque essere fermata dalla pressione di ESCAPE.

Il comando *BUILD crea un file contenente righe di testo introdotte da tastiera, che in seguito potranno essere eseguite con *EXEC. Digitando:

*BUILD droide RETURN

il computer visualizzerà:

0001

e memorizzerà quanto battuto nel file 'Droide' (senza aggiungere numeri alle righe, che vengono visualizzati semplicemente come riferimento). Al termine di ogni riga occorre premere RETURN. Per indicare al computer che si è terminata l'introduzione di una riga, premere RETURN e ESCAPE al termine dell'introduzione. Come si sarà notato, il comando *LIST è l'esatto contrario di questo comando.

Il comando *BUILD rappresenta un semplice modo per memorizzare sequenze di comandi che vengono utilizzate frequentemente, ad esempio per definire il contenuto dei tasti funzione. Notare che ogni riga di testo preceduta da '*|' viene ignorata dal sistema operativo. Ad esempio:

*BUILD tasti RETURN
0001 *|File di definizione tasto funzione RETURN
0002 *KEY 0 OLD|MLIST|M RETURN
0003 *KEY 1 NEW|MAUTO|M RETURN
ESCAPE

Un comando simile è *APPEND, che permette di aggiungere un nuovo testo alla fine di un file esistente. Questo comando viene utilizzato esattamente come *BUILD, con l'eccezione che il file specificato *deve* necessariamente esistere, in caso contrario viene visualizzato il messaggio 'Not found' (non trovato).

Altri comandi

Il PC 128 S dispone di diversi altri comandi, chiamati programmi di utilità: *DUMP, *CREATE, *CLOSE e *HELP.

*DUMP deve essere seguito da un nome di file e genera un listato dei byte del file in esadecimale e ASCII. Se il file è molto lungo, queste informazioni vengono fatte scorrere sullo schermo fino a scomparire. Per arrestare la visualizzazione premere SHIFT + CTRL. Per fermare la visualizzazione può essere utilizzato anche il tasto ESCAPE.

*CREATE crea un file vuoto della lunghezza specificata. Le applicazioni di questo comando sono piuttosto limitate. *CREATE viene utilizzato come segue:

*CREATE pasticcio 1000 RETURN

Con questa riga si crea un file denominato 'Pasticcio', con lunghezza &1000 byte.

*CLOSE provoca la chiusura da parte del PC 128 S di tutti i file aperti dal sistema di archiviazione corrente. Questo comando viene generalmente utilizzato solo alla fine di un programma o di una sessione di programmazione, oppure dopo la visualizzazione del messaggio 'Already open' (già aperto).

Infine, *HELP fornisce informazioni circa i comandi utilizzati da diverse parti di un programma. Ad esempio:

*HELP ADFS RETURN

creerà una lista dei comandi utilizzabili da ADFS insieme alla loro sintassi, mentre:

*HELP . RETURN

creerà una lista simile per *tutti* i programmi ROM nel computer. In questo modo si potrà fare a meno di consultare tutte le volte il manuale.

7. Espansione del sistema

Questa sezione descrive i dispositivi che possono essere utilizzati per ampliare le prestazioni del computer base, cioè le **periferiche** (stampanti, joystick, ecc.). Lo scopo di questa parte del manuale è di fornire informazioni di base, che possono essere sviluppate con la lettura della documentazione fornita con ogni periferica o unità di espansione.

Le periferiche vengono collegate alle prese sul pannello posteriore del computer. Inoltre, sul lato destro del computer è presente un connettore di espansione che fornisce tutti i segnali elettrici necessari per il collegamento di dispositivi più sofisticati.

Collegamento di un monitor a colori e di un monitor monocromatico

Il computer è stato acquistato unitamente ad un dispositivo di visualizzazione, chiamato **monitor**, con video a colori o monocromatico. I monitor a colori permettono di sfruttare le caratteristiche grafiche messe a disposizione dal computer PC 128 S e visualizzano la gamma completa degli otto colori fissi e degli otto colori lampeggianti. I monitor monocromatici visualizzano l'immagine generata dal computer in un colore fondamentale e in diverse sfumature del colore stesso, generalmente su sfondo nero. I monitor monocromatici in genere visualizzano l'immagine in bianco, verde o ambra. Questi ultimi due colori vengono impiegati in quanto affaticano meno la vista rispetto agli altri colori. I monitor monocromatici vengono utilizzati comunemente per operazioni di word processing dove l'operatore deve passare molto tempo guardando lo schermo.

Oltre che all'alimentazione, il monitor deve essere collegato al computer per mezzo di un cavo fornito con il monitor stesso. I monitor a colori devono essere collegati al connettore 'RGB' sul pannello posteriore del computer, mentre la maggior parte dei monitor monocromatici vengono collegati al connettore 'Video Out' e solo alcuni al connettore 'RGB'. Per ulteriori informazioni consultare la documentazione fornita con il monitor.

Collegamento di un televisore

Il computer è fornito di un adattatore opzionale per il collegamento ad un normale apparecchio televisivo. Questo dispositivo viene alimentato dal computer e possiede due cavi, uno dei quali va collegato al connettore 'RGB' mentre l'altro va inserito nella presa dell'antenna del televisore. Selezionare un canale libero con il controllo di sintonia del televisore e, con il computer acceso, regolarlo finché non appare l'immagine nella parte superiore sinistra del video.

Questa opzione è particolarmente indicata per coloro che, già in possesso di una configurazione con monitor monocromatico, volessero avere a costi contenutissimi la possibilità di visualizzare immagini a colori senza acquistare subito anche un monitor a colori.

Disk drive

Disk drive supplementare per dischetti 3,5"

ADFS può supportare un secondo disk drive per dischetti da 3,5". Questo drive può essere installato sul lato destro dell'unità a disco, accanto al disk drive esistente. Il secondo drive permette di accedere ad una quantità doppia di informazioni (massimo 1280 Kbyte) senza dover scambiare i dischi ed inoltre semplifica molte operazioni di copiatura eliminando lo scambio dei dischi.

Questa installazione è un lavoro piuttosto complesso e, a meno che non si abbia esperienza al riguardo, si consiglia di contattare il rivenditore.

Collegamento di un disk drive esterno per dischetti da 5,25"

Per poter accedere ai dati in un dischetto da 5,25" occorre installare un disk drive esterno, collegandolo tramite un adattatore che si collega in linea con il cavo del disk drive normale.

Collegamento di una stampante

Se si utilizza il computer per word processing o altri lavori di ufficio la stampante è indispensabile ed è inoltre molto utile se si scrivono programmi propri.

Il computer può essere utilizzato con la maggior parte delle stampanti disponibili sul mercato, che vengono collegate al computer in uno dei due modi seguenti:

- tramite un'**interfaccia parallela**, conosciuta come 'tipo–Centronics';
- tramite un'**interfaccia seriale** utilizzando l'interfaccia standard 'RS232–C' o 'V–24'.

Tutte le stampanti vengono collegate all'alimentazione di rete. Le stampanti parallele si collegano al connettore 'Printer' situato sul retro del computer, mentre le stampanti seriali si collegano al connettore opzionale RS232, anch'esso sul retro del computer. Il connettore RS232 può ricevere una spina a 5 pin tipo 'Domino' secondo norme DIN, quindi sarà necessario un cavo di interconnessione con questa spina ad un'estremità e una spina adatta al connettore della stampante all'altra estremità. Per i cavi di stampante rivolgersi al rivenditore.

E' necessario comunicare al computer, per mezzo del comando *FX5, la stampante da utilizzare:

- *FX5,1 RETURN indica al computer che si utilizzerà una stampante parallela;
- *FX5,2 RETURN indica al computer che si utilizzerà una stampante seriale. In questo caso occorre impostare anche la velocità (baud rate) a cui funziona l'interfaccia RS232, per mezzo del comando *FX8. Per informazioni sul valore di default di baud rate per la propria stampante, consultare la documentazione fornita con la stampante stessa.

I valori di baud rate vengono selezionati come segue:

*FX8,1 RETURN – 75 baud
*FX8,2 RETURN – 150 baud
*FX8,3 RETURN – 300 baud
*FX8,4 RETURN – 1200 baud
*FX8,5 RETURN – 2400 baud
*FX8,6 RETURN – 4800 baud
*FX8,7 RETURN – 9600 baud
*FX8,8 RETURN – 19200 baud (non garantiti)

Molte stampanti incorporano una funzione di avanzamento riga automatico che potrebbe provocare la stampa di righe vuote. Per eliminare il problema ordinare al computer di non inviare il carattere di avanzamento riga (codice ASCII 10), per mezzo del comando:

*FX6,10 RETURN

Controllo della stampa

Digitando il comando BASIC VDU 2 si ordina al computer di copiare tutti gli output inviati ai driver VDU (e allo schermo). Un successivo comando VDU 3 riporterà l'output solo allo schermo.

Ad esempio, per listare sulla stampante un programma BASIC che si trova nella memoria, battere:

VDU 2 RETURN
LIST RETURN
VDU 3 RETURN

Se, per una qualsiasi ragione, la stampante non fosse pronta a ricevere le informazioni (ad es. non è collegata o è semplicemente spenta), le spie dei tasti 'Caps Lock' e 'Shift Lock' si illumineranno. Se per il momento non fosse possibile approntare la stampante, premere ESCAPE e ripetere più tardi il comando VDU 2.

Il word processor VIEW fornisce funzioni per il controllo della stampa, compresi un comando 'PRINT' e dei 'codici di evidenziazione' che possono essere introdotti nel testo. Queste funzioni sono state discusse precedentemente nella sezione riguardante VIEW.

L'interfaccia RS232

L'interfaccia opzionale RS232 serve a molti scopi oltre che a controllare le stampanti come descritto in precedenza. 'RS232' è uno standard elettrico comune a molti tipi di computer e periferiche. Il **MODEM**, un dispositivo che permette al computer di inviare i dati via linea telefonica, è un comune dispositivo RS232, così come lo è il **plotter parallelo**, che stampa immagini grafiche.

La presa RS232 riceve una spina a 5 pin tipo 'Domino' secondo norme DIN, la cui configurazione dei pin viene mostrata nell'Appendice 1. Per l'acquisto dell'interfaccia e dei dispositivi RS232 rivolgersi al rivenditore.

Collegamento di un joystick, mouse o trackball

Molte applicazioni richiedono input bidimensionali che offrano maggiore precisione dei tasti di movimento cursore. In questi casi viene utilizzato un dispositivo di puntamento quale:

- un **joystick**;
- un **mouse**, un dispositivo che si sposta con la mano con facilità;
- un **trackball**, che è simile al mouse ma possiede una sfera nella parte superiore che viene utilizzata per lo spostamento.

Tutti questi dispositivi di puntamento vengono collegati al connettore 'Joystick/Mouse' sul pannello posteriore. Questo connettore è di tipo D a 9 pin e può essere utilizzato per il collegamento di qualsiasi joystick compatibile Atari e per la maggior parte dei dispositivi conformi a questo standard.

La rete Econet

Il computer può agire da terminale su una rete locale Econet. La rete Econet può interconnettere fino ad un massimo di 250 computer, permettendo la condivisione di risorse costose quali disk drive fissi e stampanti di alta qualità. La rete Econet permette inoltre di collegare microcomputer a reti commerciali più veloci e quindi a servizi di trasmissione dati internazionali.

L'interfaccia Econet viene fornita come piastra ad innesto da inserire nella tastiera. Il collegamento viene effettuato al cavo della rete per mezzo di un cavo volante collegato alla presa 'Econet' situata sul pannello posteriore. Per ulteriori informazioni consultare la documentazione relativa all'interfaccia.

Il rivenditore potrà fornire ulteriori dettagli sui vantaggi offerti dalla rete Econet e potrà fornire interfacce ed altri prodotti di supporto.

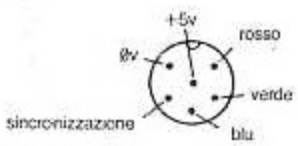
Appendice 1

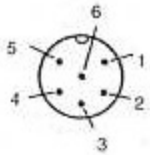
Informazioni tecniche

Configurazione dei pin dei connettori

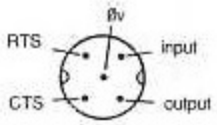
Le configurazioni dei pin per i connettori che si trovano sul retro del computer vengono illustrate nel diagramma seguente. La vista è frontale (dall'esterno del computer).

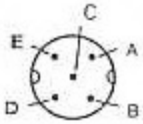
RGB



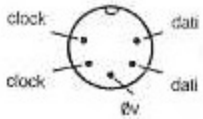


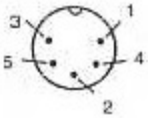
RS232



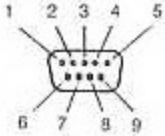


ECONET



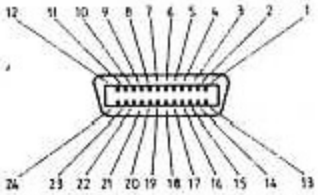


JOYSTICK



Joystick	Connettore tipo D 9 poli Collegamenti interni 6522:
1 Su (VERO= negativo)	PB3
2 Giù (VERO= negativo)	PB2
3 Sinistra (VERO= negativo)	PB1
4 Destra (VERO= negativo)	PB4
5 Collegamento interno	CB1
6 Fuoco	PB0
7 +5V	
8 0V	
9 Collegamento interno	CB2

STAMPANTE

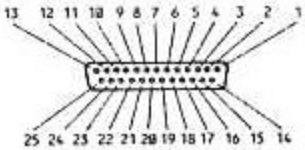


Stampante – Connettore a 24 poli

1 – Strobe (VERO= negativo)
2 – Dati 0
3 – Dati 1
4 – Dati 2
5 – Dati 3
6 – Dati 4
7 – Dati 5
8 – Dati 6
9 – Dati 7
10 – Riconoscimento (VERO= negativo)
11 – Nessun collegamento
12 – Nessun collegamento

13 – Terra
14 – Terra
15 – Terra
16 – Terra
17 – Terra
18 – Terra
19 – Terra
20 – Terra
21 – Terra
22 – Terra
23 – Terra
24 – Terra

DRIVE

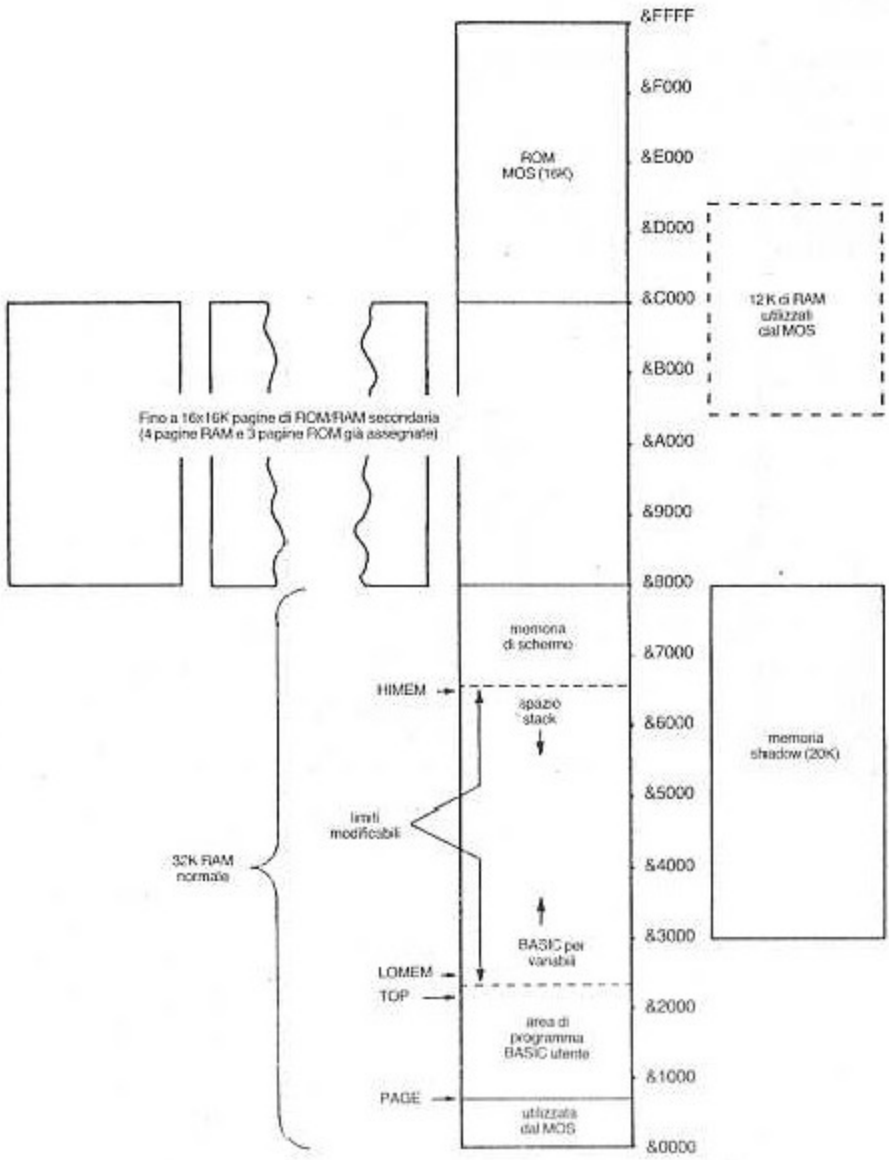


Drive – Presa tipo D 25 poli

1 Indice
2 Selezione drive 0
3 Selezione drive 1
4 Nessun collegamento
5 Controllo testina motore
6 Direzione
7 Ricerca/Step
8 Scrittura dati
9 Abilitazione scrittura
10 Traccia 0
11 Protezione in scrittura
12 Lettura dati
13 Selezione faccia

14 – Terra
15 – Terra
16 – Terra
17 – Terra
18 – Terra
19 – Terra
20 – Terra
21 – Terra
22 – Terra
23 – Terra
24 – Terra
25 – Terra

Mappa di memoria



Appendice 2

Caratteristiche dei modi

La Tabella 1 fornisce le caratteristiche di testo, set di caratteri, immagini e colore per ciascuno degli otto modi di visualizzazione standard ed i corrispondenti modi shadow.

Le informazioni relative alle assegnazioni dei colori di default per i modi da 0 a 6 ed i corrispondenti modi shadow vengono fornite nella Tabella 2. Il cambio di modo provoca sempre la selezione di un primo piano bianco o verde e di uno sfondo nero per testo e immagini (se disponibili).

Tabella 1

Modo	Righe di testo	Colonne di testo	Set di caratteri	Pixel grafici	Colori
0(128)	32	80	ASCII	640 x 256	2
1(129)	32	40	ASCII	320 x 256	4
2(130)	32	20	ASCII	160 x 256	16
3(131)	25	80	ASCII	–	2
4(132)	32	40	ASCII	320 x 256	2
5(133)	32	20	ASCII	160 x 256	4
6(134)	25	40	ASCII	–	2
7(135)	25	40	TELETEXT	(Vedere App. 3)	

Tabella 2

Modo	Primo piano	Sfondo	Colore
0(128)			
3(131)	0	128	Nero
4(132)	1	129	Bianco
6(134)			
	0	128	Nero
1(129)	1	129	Rosso
5(133)	2	130	Giallo
	3	131	Bianco
	0	128	Nero
	1	129	Rosso
	2	130	Verde
	3	131	Giallo
	4	132	Blu
	5	133	Magenta
	6	134	Azzurro
2(130)	7	135	Bianco
	8	136	*Nero/Bianco
	9	137	*Rosso/Azzurro
	10	138	*Verde/Magenta
	11	139	*Giallo/Blu
	12	140	*Blu/Giallo
	13	141	*Magenta/Verde
	14	142	*Azzurro/Rosso
	15	143	*Bianco/Nero

*coppia di colori lampeggianti

Appendice 3

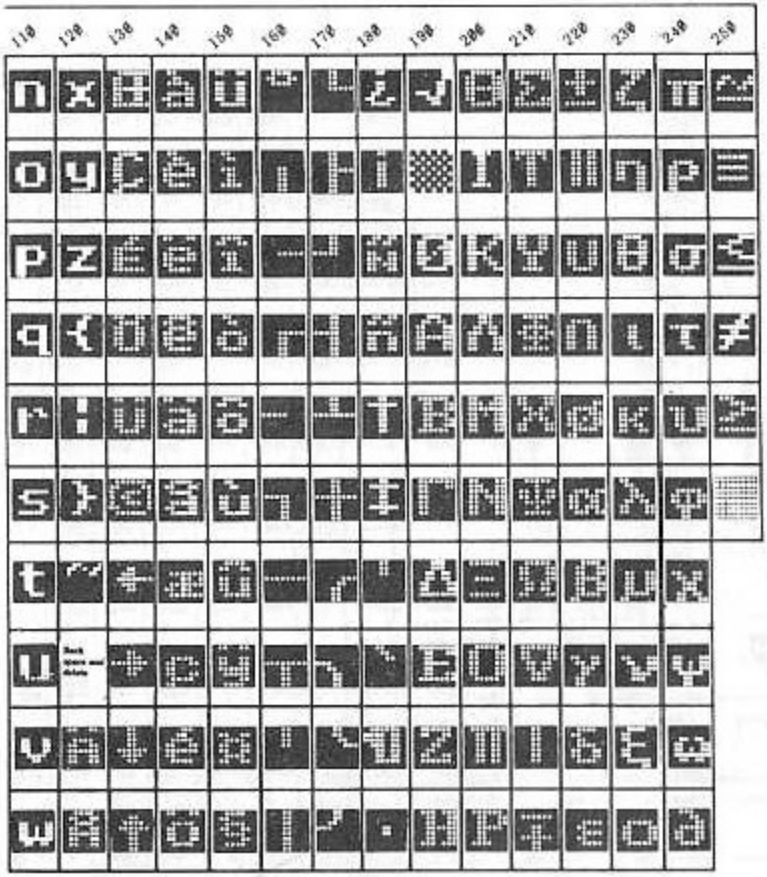
Set di caratteri

Set di caratteri ASCII visualizzati (modi da 0 a 6 e da 128 a 134)

I codici ASCII da 0 a 31 sono codici di controllo che corrispondono ai codici VDU descritti nell'Appendice 6.

	0	10	20	30	40	50	60	70	80	90	100
0	Niente	Giu	Colori logici di default	Spostamento cursore testo su 00	(	2	<	F	P	Z	d
1	Invia a stampante	Su	Disattivazione VDU	Spostamento cursore testo	)	3	=	G	Q	L	e
2	Avvio stampante	Cancellazione testo	Selezione modo	■	*	4	>	H	R	\	f
3	Arresto stampante	Inizio riga	Riprogrammazione caratteri	!	+	5	?	I	S	]	g
4	Separazione cursori	Modo pagina	Definizione area grafica	"	,	6	@	J	T	^	h
5	Unione cursori	Modo di scorrimento	Tracciamento	#	-	7	A	K	U	_	i
6	Attivazione VDU	Cancellazione grafica	Area di testo/grafica di default	$	.	8	B	L	V	£	j
7	Segnale acustico	Definizione area testo	Niente	%	/	9	C	M	W	a	k
8	Indietro	Definizione colore grafica	Definizione colore testo	&	0	:	D	N	X	b	l
9	Avanti	Definizione colori logici	Definizione origine immagine	'	1	;	E	O	Y	c	m

Ogni carattere visualizzato è composto da otto righe di otto punti.



Set di caratteri alfanumerici Teletext visualizzati (modi 7 e 135)

I codici compresi nella gamma da 0 a 31 sono codici di controllo che corrispondono ai codici VDU descritti nell'Appendice 6.

I codici nella gamma da 128 a 159 sono i codici di controllo Teletext che agiscono sui caratteri successivi sulla stessa riga. I caratteri mostrati nella tabella sono quelli visualizzati utilizzando il codice di controllo alfanumerico.

	0	10	20	30	40	50	60	70	80	90	100	110	120
0	Niente	Giu	Niente	Spostamento cursore su 00									
1	Invio a stampante	Su	Disattivazione VDU	Spostamento cursore									
2	Avvio stampante	Cancellazione schermo	Selezione modo										
3	Arresto stampante	Inizio riga	Riprogrammazione caratteri										
4	Niente	Modo pagina	Niente										
5	Niente	Modo di scorrimento	Niente										
6	Attivazione VDU	Niente	Niente										
7	Segnale acustico	Niente	Niente										Backspace e DEL
8	Indietro	Niente	Niente										Niente
9	Avanti	Niente	Niente										Alfanumerico rosso

Ciascun codice produce un carattere unico. VDU 78 o PRINT CHR$(78) visualizzeranno una N in quanto la colonna 70, riga 8 visualizza una N.

130	140	150	160	170	180	190	200	210	220	230	240	250
Alfanumerico verde	Altezza normale*	Grafico azzurro										
Alfanumerico giallo	Doppia altezza	Grafico bianco										
Alfanumerico blu	Niente	Nasconde visualizzazione										
Alfanumerico magenta	Niente	Grafica contigua*										
Alfanumerico azzurro	Niente separata*	Grafica										
Alfanumerico bianco*	Grafico rosso	Niente										
Lampeggiamento	Grafico verde	Sfondo nero*										
Immobile*	Grafico giallo	Nuovo sfondo										
Niente	Grafico blu	Blocco grafica										
Niente	Grafico magenta	Rilascio grafica*										

*ogni riga inizia con queste opzioni

Caratteri grafici Teletext visualizzati

I codici compresi nella gamma da 0 a 31 sono i codici di controllo che corrispondono ai codici VDU descritti nell'Appendice 6.

I codici nella gamma da 128 a 159 cono i codici di controllo Teletext che agiscono sui successivi caratteri nella stessa riga. I caratteri mostrati nella tabella sono quelli visualizzati utilizzando il codice di controllo grafico.

	0	10	20	30	40	50	60	70	80	90	100	110	120
0	Niente	Giu	Niente	Spostamento cursore su 00									
1	Invio a stampante	Su	Disattivazione VDU	Spostamento cursore									
2	Avvio stampante	Cancellazione schermo	Selezione modo										
3	Arresto stampante	Inizio riga	Riprogrammazione caratteri										
4	Niente	Modo pagina	Niente										
5	Niente	Modo di scorrimento	Niente										
6	Attivazione VDU	Niente	Niente										
7	Segnale acustico	Niente	Niente										Backspace e DEL
8	Indietro	Niente	Niente										Niente
9	Avanti	Niente	Niente										Alfanumerico rosso

Ciascun carattere possiede un codice. H corrisponde al codice 72 in quanto la sua locazione è colonna 70, riga 2.

130	140	150	160	170	180	190	200	210	220	230	240	250
Alfanumerico verde	Altezza normale*	Grafico azzurro										
Alfanumerico giallo	Doppia altezza	Grafico bianco										
Alfanumerico blu	Niente	Nascondie visualizzazione										
Alfanumerico magenta	Niente	Grafica contigua*										
Alfanumerico azzurro	Niente separata*	Grafica										
Alfanumerico bianco*	Grafico rosso	Niente										
Lampeggiamento	Grafico verde	Sfondo nero*										
Immobile*	Grafico giallo	Nuovo sfondo										
Niente	Grafico blu	Blocco grafica										
Niente	Grafico magenta	Rilascio grafica*										

*ogni riga inizia con queste opzioni.

Codici della tastiera

Il codice prodotto da ciascun tasto della tastiera dipende dalle impostazioni di *caps lock*, *shift lock* e dalla pressione di SHIFT o CTRL. Per ciascun tasto nella figura riportata di seguito:

- il numero più basso è il codice per i caratteri prodotti dalla semplice pressione del tasto corrispondente;
- il numero mediano è il codice per i caratteri prodotti con la pressione di SHIFT;
- il numero più alto è il codice generato quando il tasto viene premuto insieme a CTRL.

I codici generati dai 10 tasti funzione possono essere specificati dall'utente (vedere Appendice 5).

I tasti di movimento cursore producono dei codici solamente se sono attivati da *FX4 (vedere Appendice 5).

Tutti i numeri sono espressi in notazione decimale.

																	139 139 139	43 43 43	45 45 45	47 47 47	42 42 42
27 27 27	49 33 49	50 34 50	51 35 51	52 36 52	53 37 53	54 38 54	55 39 55	56 40 56	57 41 57	0 64 48	45 61 45	30 126 94	28 124 92	136 136 136	137 137 137	55 55 55	56 56 56	57 57 57	35 35 35
9 9 9	17 81 113	23 87 119	5 69 101	18 82 114	20 84 116	25 89 121	21 85 117	9 73 105	15 79 111	16 80 112		27 123 91	31 96 95	138 138 138		52 52 52	53 53 53	54 54 54	127 127 127
CAPS LOCK	CTRL	1 65 97	19 83 115	4 68 100	6 70 102	7 71 103	8 72 104	10 74 106	11 75 107	12 76 108	59 43 59	58 42 58	29 125 93	13 13 13		49 49 49	50 50 50	51 51 51	44 44 44
SHIFT LOCK	SHIFT	26 90 122	24 88 120	3 67 99	22 86 118	2 66 98	14 78 110	13 77 109	44 60 44	46 62 46	47 63 47	SHIFT		127 127 127	135 135 135	48 48 48	46 46 46	13 13 13	
			32 32 32																

Appendice 4

Comandi del sistema operativo

I comandi del sistema operativo permettono di comunicare le richieste al MOS. Di seguito viene riportata una breve descrizione di questi comandi. Nell'Appendice 7 è contenuto un riassunto dei comandi applicabili ai due sistemi di archiviazione.

I comandi del sistema operativo vengono introdotti direttamente da tastiera (in questo caso sono sempre seguiti dalla pressione del tasto RETURN), oppure vengono incorporati all'interno di programmi (nei quali devono comparire come ultima istruzione di una riga).

Quasi tutti i comandi possono essere abbreviati utilizzando i primi caratteri del comando seguiti da un ".". L'abbreviazione minima utilizzabile viene fornita tra parentesi accanto ad ogni nome di comando.

I comandi contrassegnati dal simbolo □ vengono applicati a funzioni che possono essere utilizzate da utenti esperti. Per dettagli su questi comandi consultare il manuale di riferimento.

*CODE □ Permette di eseguire routine in linguaggio macchina che si trovano già in memoria come se fossero parte del MOS.

*CONFIGURE(*CO.) Fornisce un mezzo diretto per modificare le impostazioni nella EEPROM (senza utilizzare il programma di utilità Control Panel descritto a pag. 34). *CONFIGURE accetta uno o due parametri. Il primo parametro è il nome dell'impostazione da modificare, il secondo (se necessario) è il valore da memorizzare nella EEPROM. Notare che è possibile utilizzare i valori di default tenendo premuto il tasto 'R' mentre si accende il computer. Nella lista di parametri qui sotto n e m stanno per un numero decimale e x per un numero in notazione esadecimale.

□	BAUD *n*	Modifica la velocità di tramissione/ricezione in base al valore *n*.
	BOOT	Inverte le azioni di BREAK e SHIFT + BREAK.
	CAPS	Imposta l'opzione CAPS LOCK.
□	DATA *n*	Modifica l'impostazione del formato dei dati RS232 in base al valore *n* introdotto.
	DELAY *n*	Modifica l'impostazione dell'intervallo di auto– ripetizione della tastiera a *n* centesimi di secondo.
□	DIR	Inizializza ADFS con la directory selezionata.

□ FDRIVE *n*	Configura il controllore del disco per diversi tipi di unità a disco in base al valore *n*.
FILE *x*	Modifica l'impostazione di default del sistema di archiviazione nell'impostazione contenuta nella presa ROM *x*.
IGNORE *n*	Modifica il 'carattere di riconoscimento stampante' nel carattere ASCII corrispondente a *n*. Omettendo *n* tutti i caratteri vengono inviati alla stampante.
LANG *x*	Modifica l'impostazione di default del linguaggio nell'impostazione contenuta nella presa ROM *x*.
LOUD	Porta al massimo il volume della segnalazione acustica.
MODE *n*	Imposta la visualizzazione su n (da 0 a 7 o da 128 a 135).
NOBOOT	Assegna la funzione normale a BREAK e SHIFT + BREAK.
NOCAPS	Ripristina l'opzione CAPS LOCK.
□ NODIR	Inizializza ADFS senza una directory selezionata.
NOSCROLL	Attiva l'opzione di protezione dello scorrimento.
PRINT *n*	Modifica l'impostazione del tipo di stampante in base al valore *n* (vedere pag. 176).
PROPORTIONAL	Configura la simulazione tasto cursore per il joystick per produrre una gamma completa di valori.
QUIET	Porta a metà volume l'impostazione della segnalazione acustica.
REPEAT *n*	Modifica l'impostazione dell'intervallo di auto–ripetizione dei tasti a n centesimi di secondo.
SCROLL	Disattiva l'opzione di protezione scorrimento.
SHCAPS	Imposta l'opzione SHIFT + CAPS LOCK.
STICK *n*	Imposta a *n* l'accelerazione del joystick nel modo PROPORZIONALE (da 0 a 7).
SWITCHED	Configura la simulazione tasto cursore del joystick per produrre una gamma limitata di valori.

TV *n,m* — Imposta l'allineamento verticale e l'opzione di visualizzazione del doppio delle righe (interlace) (come per il comando *TV descritto più avanti).

Quando viene utilizzato senza parametro, *CONFIGURE visualizza una lista delle opzioni riportate qui sopra.

*GO

□ Esegue un programma in linguaggio macchina nella memoria centrale del computer.

*HELP (*H.)

Visualizza dati riguardo al PC 128 S, ai linguaggi e ai sistemi di archiviazione residenti al momento (ma non necessariamente selezionati) nel computer.

*IGNORE (*IG.)

Imposta il 'carattere di riconoscimento stampante' (vedere pag. 176). Ad esempio:

*IGNORE 10 — la stampante ignora i caratteri di avanzamento riga (ASCII = 10).

*INSERT (*INS.)

□ E' possibile fare in modo che il MOS ignori la presenza di una data ROM utilizzando il comando *UNPLUG descritto qui sotto. *INSERT riporta allo stato normale la ROM di una presa specifica, solo dopo un'interruzione hard o una successiva accensione.

*KEY (*K.)

Assegna una sequenza di caratteri ad un tasto funzione specifico (vedere pag. 48). Ad esempio:

*KEY 0 assegna il comando BASIC LIST
LIST | M al tasto funzione 0.

*LINE

□ Questo comando è simile al comando *CODE, con la differenza che fornisce un mezzo per passare una riga di testo (cioè la parte rimanente del programma) alla routine.

*MOVE

□ Copia i file da un sistema di archiviazione all'altro, ad esempio:

*MOVE –ROM–INDIRIZZO –ADFS–BACKUP

copia il file INDIRIZZO dalla ROM ad un disco ADFS, assegnandogli il nome BACKUP.

*ROMS

Visualizza una lista delle ROM presenti al momento in una delle prese ROM e nelle pagine RAM secondaria. Il listato fornisce il numero di presa (in esadecimale), il nome della ROM e indica se la ROM è disponibile (vedere *UNPLUG più avanti).

*SHADOW (*SH.)

Attiva la memoria principale o la memoria shadow:

*SHADOW 0 provoca la selezione della memoria shadow in base a successive modifiche di modo (anche se il modo selezionato fosse nella gamma da 0 a 7);

*SHADOW 1 provoca la selezione della memoria principale o della memoria shadow in base al numero del modo (i modi da 0 a 7 selezionano la memoria principale, i modi da 128 a 135 selezionano la memoria shadow).

*SHOW

Visualizza la sequenza di caratteri assegnata correntemente ad un tasto funzione specifico. Ad esempio:

*SHOW 0 visualizza la sequenza di caratteri assegnata correntemente al tasto funzione 0.

*SHUT

Chiude tutti i file aperti correntemente riconosciuti dal PC 128 S, independentemente dal sistema di archiviazione corrente.

*SRDATA

□ Riserva un'area designata di RAM secondaria da utilizzare con i dati.

*SRREAD

□ Copia un'area designata di RAM secondaria sulla RAM normale.

*SRROM

□ Riserva un'area designata di RAM secondaria per l'utilizzo con un indirizzamento assoluto.

*SRWRITE

□ Copia un'area designata di RAM normale sulla RAM secondaria.

*STATUS (*ST.)

Visualizza il contenuto corrente delle diverse impostazioni nella EEPROM. *STATUS può inoltre essere utilizzato con uno dei parametri descritti in *CONFIGURE. In questo caso viene visualizzata solo l'impostazione specificata. Ad esempio:

*STATUS DELAY visualizza l'impostazione dell'intervallo di auto–ripetizione della tastiera nella EEPROM;

*STATUS visualizza tutte le impostazioni della EEPROM.

*TIME

Su questo computer questo comando ritorna il valore internazionale assurdo "Fri,31 Dec 1999.23:59:59".

*TV

Vengono utilizzati due parametri, il primo indica una regolazione all'allineamento verticale, il secondo definisce l'impostazione dell'opzione interlace. Ad esempio:

*TV0,1	non provoca alcun allineamento verticale ma disattiva l'interlace;
*TV1,0	provoca lo scorrimento dello schermo di una riga verso l'alto e attiva l'interlace;
*TV255,1	provoca lo scorrimento dello schermo di una riga verso il basso e disattiva l'interlace.

*UNPLUG (*UNP.)

□ Il PC 128 S ignora la ROM in una presa specifica dopo la prossima interruzione hard o accensione.

Appendice 5

Comandi *FX

Una parte della memoria riservata per l'utilizzo da parte del PC 128 S viene riservata alla memorizzazione di informazioni relative allo stato corrente della macchina e alle sue reazioni dettate da diverse circostanze. Le informazioni sono accessibili direttamente e possono essere modificate dall'utente per mezzo di richiami a sistemi operativi (chiamati anche richiami **OSBYTE**).

Alcuni richiami OSBYTE hanno un comando *FX equivalente che può essere introdotto direttamente da tastiera oppure incluso in un programma BASIC. I comandi *FX sono riassunti di seguito. I comandi contrassegnati dal simbolo □ permettono di accedere a funzioni che verranno utilizzate solo dagli utenti esperti. Per ulteriori informazioni su questi comandi consultare il manuale di riferimento.

Nella sequenza sono presenti dei salti di numero che sono dovuti al fatto che alcuni richiami OSBYTE non possiedono l'equivalente comando *FX.

I parametri nei comandi *FX possono essere separati da una virgola (come mostrato negli esempi) oppure da una sequenza di uno o più spazi.

Tralasciando un parametro, viene assunto il valore di default 0.

*FX0 Visualizza la versione del PC 128 S.

*FX1 □ Riservato per programmi applicativi.

*FX2 Seleziona il flusso di input:

*FX2,0	solo tastiera (disattiva l'input RS232 (opzionale);
*FX2,1	solo input RS232 (opzionale);
*FX2,2	input tastiera e RS232 bufferizzato (opzionale).

*FX3 Seleziona il flusso di output:

*FX3,0	solo stampante e schermo;
*FX3,1	stampante, schermo e RS232 (opzionale);
*FX3,2	solo stampante;
*FX3,3	stampante e RS232 (opzionale);
*FX3,4	solo schermo;
*FX3,5	schermo e RS232 (opzionale);
*FX3,6	nessuno;
*FX3,7	solo RS232 (opzionale)

□ Possono essere utilizzati anche altri valori.

*FX4 Attiva/disattiva i tasti cursore:

*FX4,0 attiva i tasti cursore;
*FX4,1 disattiva i tasti cursore ed assegna i codici ASCII:
COPY 135
← 136
→ 137
↓ 138
↑ 139
*FX4,2 disattiva i tasti cursore ed assegna i tasti dedicati:
COPY 11
← 12
→ 13
↓ 14
↑ 15
*FX4,3 i tasti cursore influiscono sui movimenti del joystick.
COPY è equivalente al pulsante 'fuoco' (e viene eseguito un OR).

*FX5 Seleziona il tipo di stampante (vedere pag. 176):

*FX5,0 seleziona il sink di stampante (nessuna stampa);
*FX5,1 seleziona la stampante parallela;
*FX5,2 seleziona la stampante seriale;
□ *FX5,3 seleziona una routine utente di stampante;
*FX5,4 seleziona un server di stampante di rete.

I tipi di stampante con valore maggiore di 4 non devono essere utilizzati.

L'impostazione di default viene impostata con *CONFIGURE PRINT (vedere Appendice 4).

*FX6 Seleziona il carattere di riconoscimento stampante (equivalente a *IGNORE). Ad esempio:

*FX6,10 permette di non inviare alla stampante il carattere di avanzamento riga (ASCII = 10).

*FX7 □ Seleziona la velocità di ricezione dell'RS232 (opzionale).

*FX8 □ Seleziona la velocità di trasmissione dell'RS232 (opzionale).

*FX9 Imposta la velocità di lampeggiamento del primo colore in cinquantesimi di secondo (impostazione di default = 25).

*FX9,0 disattiva il lampeggiamento e forza il primo colore sullo schermo;
*FX9,10 imposta la velocità di lampeggiamento a un quinto di secondo.

*FX10 Imposta la velocità di lampeggiamento del secondo colore in cinquantesimi di secondo (impostazione di default = 25).

*FX10,0 disattiva il lampeggiamento e forza il secondo colore sullo schermo;
*FX10,5 imposta la velocità di lampeggiamento a un decimo di secondo.

*FX11 Imposta l'intervallo di auto–ripetizione dei tasti in centesimi di secondo (impostazione di default = 32, oppure come impostata da *CONFIGURE DELAY).

*FX11,0 disattiva l'auto–ripetizione;

*FX11,10 imposta l'intervallo di auto–ripetizione a un decimo di secondo.

*FX12 Imposta il periodo di auto–ripetizione dei tasti in centesimi di secondo (impostazione di default = 8 oppure come impostata da *CONFIGURE REPEAT).

*FX12,0 ripristina le impostazioni di default di intervallo di auto–ripetizione e periodo di auto–ripetizione;
*FX12,3 imposta il periodo di auto–ripetizione a tre centesimi di secondo.

*FX13 □ Disattiva diversi eventi.

*FX14 □ Attiva diversi eventi.

*FX15 Scarica i buffer:

*FX15,0 scarica tutti i buffer;
*FX15,1 scarica il buffer di input corrente.

*FX18 Cancella le definizioni di tasti funzione eseguite dall'utente.

*FX19 □ Attende la sincronizzazione verticale.

*FX20 Ripristina le definizioni di font di default, cioè riporta alla condizione normale i caratteri corrispondenti ai codici ASCII da 32 a 126.

*FX21 Scarica il buffer selezionato:

*FX21,0 buffer della tastiera;
*FX21,1 buffer di input dell'RS232 (opzionale);
*FX21,2 buffer di output dell'RS232 (opzionale);
*FX21,3 buffer della stampante;
*FX21,4 canale sonoro 0;
*FX21,5 canale sonoro 1;
*FX21,6 canale sonoro 2;
*FX21,7 canale sonoro 3.

*FX22 □ Incrementa il semaforo di polling della ROM.

*FX23 □ Decrementa il semaforo di polling della ROM.

*FX25 Ripristina un gruppo di definizioni di font:

*FX25,0	ripristina i codici di carattere tra 32 e 255;
*FX25,1	ripristina i codici di carattere tra 32 e 63;
*FX25,2	ripristina i codici di carattere tra 64 e 95;
*FX25,3	ripristina i codici di carattere tra 96 e 127;
*FX25,4	ripristina i codici di carattere tra 128 e 159;
*FX25,5	ripristina i codici di carattere tra 160 e 191;
*FX25,6	ripristina i codici di carattere tra 192 e 223;
*FX25,7	ripristina i codici di carattere tra 224 e 255.

*FX108 Commuta la memoria principale/shadow nella mappa centrale:

*FX108,0	commuta la memoria shadow nella mappa centrale (immediato);
*FX108,1	commuta la memoria principale nella mappa centrale.

*FX109 □ Rende permanente il sistema di archiviazione temporaneo.

*FX112 Seleziona la memoria su cui i caratteri verranno scritti finché non si modifica il modo.

*FX112,0	scrive sulla memoria specificata dalla modifica di modo;
*FX112,1	scrive sulla memoria principale (immediato);
*FX112,2	scrive sulla memoria shadow (immediato).

*FX113 Seleziona la memoria da visualizzare fino alla prossima modifica di modo:

*FX113,0	visualizza la memoria specificata dalla modifica di modo;
*FX113,1	visualizza la memoria principale (immediato);
*FX113,2	visualizza la memoria shadow (immediato).

*FX114 Seleziona la memoria principale/shadow nelle modifiche di modo successive (equivalente a *SHADOW):

*FX114,0	forza la selezione della memoria shadow;
*FX114,1	seleziona la memoria principale/shadow a seconda del numero di modo.

*FX118 □ Riflette lo stato della tastiera nei LED della tastiera.

*FX119 □ Chiude i file *SPOOL/*SPOOLON/*EXEC.

*FX120 □ Scrive informazioni sui tasti premuti.

*FX124 □ Riconosce la condizione escape senza effetti secondari.

*FX125 □ Imposta la condizione escape.

*FX126 □ Riconosce la condizione escape con effetti secondari.

*FX136 □ Definisce il punto di introduzione per la routine di utente PC 128 S (equivalente a *CODE).

*FX138 Inserisce nel buffer il codice di carattere (vedere *FX21 per una lista completa dei numeri di buffer). Ad esempio:

*FX138,0,65 inserisce ASCII 65 (A) nel buffer della tastiera.

*FX139 □ Seleziona il valore dell'opzione (equivalente a *OPT).

*FX141 Seleziona il sistema di archiviazione ROM (equivalente a * ROM).

*FX142 □ Attiva il linguaggio ROM.

*FX143 □ Emette la richiesta del serivizio ROM a pagine.

*FX144 Imposta lo scorrimento verticale dello schermo e l'opzione interlace per la successiva modifica di modo o la successiva interruzione (equivalente a *TV). Ad esempio:

*FX144,0,1 non provoca lo spostamento dello schermo e disattiva l'interlace;
*FX144,1,0 sposta lo schermo verso l'alto di una riga ed attiva l'interlace;
*FX144,255 sposta lo schermo verso il basso di una riga (ed attiva l'interlace).

*FX146–*FX151 □ Accede alle aree di I/O mappate in memoria.

*FX153 □ Inserisce nel buffer il codice di carattere, controllando presenza codici ESCAPE.

*FX154 □ Scrive sul registro di controllo Video ULA.

*FX155 □ Scrive sul registro di tavolozza Video ULA.

*FX156 □ Scrive sul registro di controllo ACIA 6850 (parte dell'interfaccia opzionale RS232).

*FX162 □ Scrive sulla EEPROM.

*FX178 □ Disattiva il controllo tastiera.

*FX179 □ Scrive il semaforo di polling ROM.

*FX180 □ OSHWM.

*FX181 □ Scrive il modo RS232 (opzionale).

*FX190 □ Scrive interpretazione ADVAL.

*FX191 □ Scrive flag utilizzo RS232 (opzionale).

*FX193 □ Scrive contatore lampeggiamento.

*FX194 □ Scrive conteggio periodo mark.

*FX195 □ Scrive conteggio periodo space.

*FX196 □ Scrive intervallo di auto–ripetizione tasti.

*FX197 □ Scrive periodo di auto–ripetizione tasti.

*FX198 □ Scrive gestione file *ECEX.

*FX199 □ Scrive gestione file *SPOOL.

*FX200 Imposta l'effetto di BREAK e ESCAPE in base a n:

*FX200,0 imposta l'azione normale di BREAK e ESCAPE;

*FX200,1 imposta l'azione normale di BREAK e disattiva ESCAPE;

*FX200,2 cancella la memoria su BREAK ed imposta l'azione normale di ESCAPE;

*FX200,3 cancella la memoria su BREAK e disattiva ESCAPE.

*FX201 □ Scrive disabilitazione tastiera.

*FX202 □ Scrive byte di stato tastiera.

*FX203 □ Scrive area handshake RS232 (opzionale)

*FX204 □ Scrive flag di soppressione input RS232 (opzionale).

*FX206– □ Utilizzato da Econet.

*FX208

*FX210 Scrive stato di soppressione suono:

*FX210,0 attiva emissione suono;

*FX210,1 disattiva emissione suono.

*FX211 Scrive il canale di segnalazione acustica (BELL) (CTRL +[G]) (impostazione di default = 3). Ad esempio:

*FX211,0 seleziona il canale 0.

*FX212 Scrive informazioni segnalazione acustica (CTRL + [G]) (impostazione di default = 144). Ad esempio:

*FX212,200 produce un suono più dolce.

*FX213 Scrive frequenza segnalazione acustica (CTRL + [G]) (impostazione di default CTRL = 101). Ad esempio:

*FX213,200 produce un suono più acuto.

*FX214 Scrive la durata della segnalazione acustica (CTRL +[G]) (impostazione di default = 7). Ad esempio:

*FX214,1 produce un suono molto breve;
*FX214,255 produce un suono infinito.

*FX215 □ Scrive soppressione messaggio di avviamento e stato opzione di caricamento (!BOOT).

*FX216 □ Scrive lunghezza della stringa tasto dedicato.

*FX217 □ Scrive numero di righe stampate dall'ultima interruzione pagina.

*FX218 □ Scrive numero di elementi nella coda VDU.

*FX219 Scrive il valore del carattere ritornato da TAB (impostazione di default = 9, cioè cursore a destra). Ad esempio:

*FX219,127 rende TAB equivalente a DELETE.

*FX220 Scrive il carattere ESCAPE (impostazione di default = 27). Ad esempio:

*FX220,32 rende [BARRA SPAZIATRICE] il tasto ESCAPE.

*FX221–*FX224 Scrive lo stato di interpretazione del codice buffer di input.

*FX225 Scrive lo stato dei tasti funzione:

*FX225,0 disattiva i tasti funzione;
*FX225,1 dà ai tasti la loro normale funzione di generazione stringhe;
*FX225,2 ritorna un codice che rappresenta il tasto preceduto da un carattere nullo.

I tasti funzione possono essere impostati anche per generare un codice ASCII singolo utilizzando *FX225,n (dove n rappresenta il numero base nella gamma da 3 a 255). Questo ha l'effetto di assegnare ASCII n a f0, ASCII n+ 1 a f1, ASCII n + 2 a f2 ecc. Ad esempio:

*FX225,65 provoca la generazione del codice ASCII 65 (A) da parte di f0, del codice ASCII 66 (B) di f1, del codice ASCII 67 (C) da parte di f2 ecc.

*FX226 Imposta il numero base per le pressioni di SHIFT + tasto funzione (impostazione di default = 128). Ad esempio:

*FX226,97 provoca la generazione del codice ASCII 97 (a) da parte di SHIFT + f0, del codice ASCII 98 (b) da parte di SHIFT + f1 ecc.

*FX227 Imposta il numero base per le pressioni di CTRL + tasto funzione (impostazione di default = 144). Ad esempio:

*FX227,48 provoca la generazione del codice ASCII 48 (0) da parte di CTRL + f0, del codice ASCII 49 (1) da parte di CTRL + f1 ecc.

*FX228 Imposta il numero base per le pressioni di SHIFT + CTRL + tasto funzione (impostazione di default = nessun effetto). Ad esempio:

*FX228,200 provoca la generazione del codice ASCII 200 da parte di SHIFT + CTRL + f0 e del codice ASCII 201 da parte di SHIFT + CTRL + f1 ecc.

*FX229 Scrive lo stato del tasto ESCAPE.

*FX229,0 dà a ESCAPE la sua normale funzione;

*FX229,1 provoca la generazione da parte di ESCAPE (o del tasto selezionato da *FX220) del proprio codice ASCII.

*FX230 □ Scrive il flag che determina gli effetti di ESCAPE.

*FX231 □ Scrive la maschera di bit IRQ per l'utente 6522.

*FX232 □ Scrive la maschera di bit IRQ per 6850 (RS232 (opzionale)).

*FX233 □ Scrive la maschera di bit di interrupt per il sistema 6522.

*FX236 □ Scrive lo stato di destinazione carattere.

*FX237 □ Scrive lo stato del cursore.

*FX238 Imposta il numero base per il tastierino numerico (impostazione di default = 48 per 0 tastierino).

*FX241 Non utilizzato.

*FX244 □ Scrive il flag di assegnazione tasto dedicato.

*FX245 □ Scrive il flag di destinazione stampante.

*FX246 □ Scrive il carattere di riconoscimento stampante.

*FX247–*FX249 □ Intercetta il vettore BREAK.

*FX254 Imposta l'effetto di SHIFT sul tastierino numerico:

*FX254,0 provoca l'attivazione di SHIFT (cioè SHIFT +0 tastierino genera !);

*FX254,1 disattiva SHIFT.

*FX255 □ Scrive le opzioni di avviamento.

Appendice 6

Codici VDU

La visualizzazione di testi e grafica è controllata da una complicata serie di routine PC 128 S denominata **driver VDU**. Il driver VDU è sempre attivo a meno che non sia stato disabilitato con *FX3 (vedere Appendice 5) o VDU 21 (vedere più avanti).

I codici descritti di seguito modificano il comportamento del driver VDU e possono essere usati per produrre una varietà di effetti. L'implementazione più comune è quella realizzata attraverso l'istruzione BASIC VDU, ma i comandi al driver VDU possono essere anche inoltrati direttamente da tastiera *premendo delle sequenze* Control (cioè la pressione simultanea di CTRL e di un altro tasto).

Alcuni codici VDU consistono di una serie di valori. Se necessario, tali valori devono essere specificati per ottenere l'effetto desiderato.

Codice	Tasto CTRL	Altri valori	Effetto
VDU 0	@	0	Nessun effetto
VDU 1	A	1	Invia il successivo carattere alla sola stampante. Per esempio: VDU 1,65 Stampa, ma non invia allo schermo il carattere A.
VDU 2	B	0	Abilita la stampante.
VDU 3	C	0	Disabilita la stampante.
VDU 4	D	0	Scrive un testo alla posizione del cursore di testo (cioè ripristina il cursore di testo e visualizza il testo seguente nelle posizioni normali dei caratteri).
VDU 5	E	0	Scrive un testo alla posizione indicata dal cursore grafico (cioè elimina il cursore di testo e visualizza il testo successivo alle coordinate grafiche). La posizione del cursore di testo rimane inalterata.
VDU 6	F	0	Riabilita l'output su schermo (cioè abilita il driver VDU).
VDU 7	G	0	Emette un segnale acustico dall'altoparlante interno.
VDU 8	H	0	Sposta di un carattere a sinistra il cursore di testo.

VDU 9	I	0	Sposta di un carattere a destra il cursore di testo.
VDU 10	J	0	Sposta il cursore di testo alla riga seguente.
VDU 11	K	0	Sposta il cursore di testo alla riga precedente.
VDU 12	L	0	Cancella lo schermo e riporta il cursore di testo alla posizione (0,0) (equivalente a CLS).
VDU 13	M	0	Sposta il cursore di testo all'inizio della riga corrente.
VDU 14	N	0	Abilita il modo pagina (cioè sospende l'output alla fine di ogni schermo pieno e attende che l'utente prema SHIFT).
VDU 15	O	0	Disabilita il modo pagina (cioè l'output non viene sospeso alla fine di ogni schermo pieno).
VDU 16	P	0	Riempie l'area grafica attualmente selezionata con il colore usato per lo sfondo (equivalente a CLG).
VDU 17	Q	1	Cambia il colore di primo piano o dello sfondo a partire dalla successiva emissione di testo (equivalente a COLOR). In modo 5 (133), per esempio: VDU 17,2 imposta il primo piano del modo testo con il colore giallo. VDU 17,129 imposta lo sfondo del modo testo con il colore rosso.
VDU 18	R	2	Cambia i colori dello sfondo o del primo piano per il successivo output grafico e definisce il modo nel quale esso deve essere visualizzato (equivalente a GCOL). In modo 2 (130), per esempio: VDU 18,0,4 imposta il colore blu come colore di primo piano per la grafica. VDU 18,0,134 imposta il colore azzurro come colore di fondo per la grafica.

VDU 19	S	5	Cambia la tavolozza dei colori. VDU 19 permette di associare uno qualsiasi dei 16 colori complessivamente disponibili ai numeri di colore effettivamente utilizzabili in un modo specificato. In modo 0 (128) per esempio: VDU 19,1,2,0,0,0 associa il colore 1 (normalmente sccociato al bianco) al verde. VDU 19,0,7,0,0,0 associa il colore 0 (di solito associato al nero) al bianco. I tre numeri al termine della sequenza di valori devono sempre essere uguali a zero.
VDU 20	T	0	Ristabilisce i colori di default (cioè riporta testo e grafica a bianco su nero) e riporta la tavolozza ai colori di default.
VDU 21	U	0	Disabilita il driver VDU (cioè inibisce tutto il successivo output allo schermo). Si noti che CTRL + U inviato da tastiera ha come effetto la cancellazione della riga corrente.
VDU 22	V	1	Seleziona il modo dello schermo. Questa sequenza non deve essere inviata da tastiera in linguaggi come il BASIC, né dai comandi di schermo della famiglia VIEW. Vedere il manuale di riferimento per ulteriori informazioni.
VDU 23	W	9	Funzione mista. VDU 23 realizza una gran quantità di funzioni, la maggior parte delle quali vanno oltre l'intento del presente manuale. Le funzioni sono elencate di seguito; i dettagli sui parametri restanti si possono trovare nel manuale di riferimento. VDU 23,0 — Controlla direttamente il 6845. VDU 23,1 — Modifica il cursore. VDU 23,2 VDU 23,3 — Imposta il riempimento pieno.

VDU 23,4	Modelli
VDU 23,5	
VDU 23,6	Seleziona linea punteggiata.
VDU 23,7	Scorrimento diretto della finestra.
VDU 23,8	Cancella un blocco nella finestra di testo.
VDU 23,9	
VDU 23,10	Imposta la velocità di lampeggiamento.
VDU 23,11	Ripristina il modello di riempimento di default.
VDU 23,12	
VDU 23,13	Imposta il modello di riempimento semplice.
VDU 23,14	Modello.
VDU 23,15	
VDU 23,16	Controlla il movimento del cursore.

Le funzioni da 17 a 31 sono riservate.

Qualsiasi valore maggiore di 31 presente in una sequenza VDU 23 è considerato come riferimento a un carattere da ridefinire.

VDU 24	X	8	Definisce la finestra grafica.
VDU 25	Y	5	Equivalente all'istruzione BASIC PLOT (vedere l'appendice 9).
VDU 26	Z	0	Ripristina la finestra grafica e di testo.
VDU 27	[	0	Nessun effetto. Notare che CTRL +[equivale a ESCAPE.
VDU 28	\	4	Definisce la finestra di testo.
VDU 29	]	4	Definisce l'origine per la grafica cioè la posizione sullo schermo a cui si attribuiscono le coordinate grafiche (0,0). Per esempio:

VDU 29,640;512; considera le successive coordinate grafiche relative al punto (640,512) all'incirca al centro dello schermo.

Notare l'uso (obbligatorio) del punto e virgola.

VDU 30	^	0	Sposta il cursore di testo a (0,0).
VDU 31	_	2	Sposta il cursore di testo alla posizione specificata (equivalente a PRINT TAB); per esempio: VDU 31,20,10 sposta il cursore alla colonna 20 della riga 10 (ricordare che la prima riga dello schermo e la prima posizione di ciascuna riga hanno indirizzo 0).
VDU 32–126			Stampa il corrispondente carattere ASCII
VDU 127		0	Sposta il cursore di una posizione a sinistra e cancella il carattere che eventualmente si trova su questa posizione (equivalente all'azione di DELETE).

Appendice 7

Comandi del sistema di archiviazione

Elenchiamo di seguito i comandi disponibili sotto i vari sistemi di archiviazione. Per la verità, molti dei comandi sono gestiti dal PC 128 S ma, a scopo di completezza, essi sono elencati (e duplicati) nella descrizione di ciascun sistema di archiviazione.

In massima parte i comandi possono essere abbreviati ai primi caratteri seguiti da un punto (.). La minima abbreviazione riconosciuta per il comando è indicata tra parentesi dopo il nome del comando.

I comandi contrassegnati da □ sono relativi ad esigenze tipiche di utenti esperti; la loro descrizione completa può essere trovata nel manuale di riferimento.

Il Sistema di archiviazione ROM

CAT (.) — Visualizza un catalogo (cioè un elenco di nomi di file accompagnati da altre informazioni) di tutte le ROM del sistema di archiviazione ROM.

*CLOSE (*CL.) — Chiude tutti i file RFS attualmente aperti.

*DUMP (*D.) — Produce una stampa dump esadecimale del file RFS specificato. Per esempio:
*DUMP MIOFILE produce un dump del file MIOFILE.

*EX □ Comando simile a *CAT (vedere sopra); a differenza del precedente, fornisce per ogni file informazioni supplementari.

*EXEC (*E.) — Dirotta l'input del MOS dalla tastiera al file RFS. Per esempio:

*EXEC START fa sì che l'input al PC 128 S provenga dal file START.

*LIST (*LI.) □ Visualizza il contenuto del file RFS specificato nel formato GSREAD con i numeri di riga.

*LOAD (*L.) □ Carica in memoria il file RFS specificato.

*OPT1 (*O.1) — Imposta il livello dell'output durante le operazioni sui file:

*OPT1,0 disattiva l'output di tutte le informazioni;
*OPT1,1 attiva l'output del nome del file, del conteggio del blocco e lunghezza;

	*OPT1,2	attiva l'output del nome del file, del conteggio del blocco, della lunghezza, dell'indirizzo di caricamento e dell'indirizzo di esecuzione.
*PRINT (*P.)	□	Produce una stampa dump in formato ASCII del file RFS specificato.
*RUN		Carica ed esegue il programma in codice macchina specificato prelevandolo dalla ROM.
*SRLOAD	□	Carica il file specificato in un'area indicata della RAM secondaria.
*TYPE (*TY.)	□	Visualizza il contenuto del file specificato in formato GSREAD senza i numeri di riga.

Il sistema avanzato di archiviazione su disco

Salvo indicazione contraria, tutti i comandi operano sulla directory attualmente selezionata.

Molti dei comandi del sistema avanzato di archiviazione su disco permettono l'uso dei caratteri wildcard * e #.

*ACCESS (*AC.)

Imposta gli attributi di un file. Gli attributi possono essere:

D Directrory
E accesso di tipo 'sola esecuzione' (solo per programmi scritti in linguaggio macchina);
L protezione file;
W accesso in scrittura;
R accesso in lettura.

Per esempio:

*ACCESS MEMO L	protegge il file MEMO;
*ACCESS PROG WR	rende disponibile il file DADSPROG in lettura e in scrittura.

*APPEND (*AP.) — Amplia il file creato tramite il comando *BUILD (vedere più avanti).

*BACK — Rende corrente la directory selezionata precedentemente.

*BACKUP — Copia un disco su un altro.

*BUILD (*BU.) — Crea un file su disco contenente una successione di righe di input.

*BYE — Chiusura di tutti i file al termine di una sessione (il suo effetto è simile a quello di *CLOSE).

CAT (.) Visualizza i nomi dei file presenti nella directory attuale o in quella specificata. Per esempio:

*CAT	visualizza il catalogo della directory attuale;
*CAT $.CASA	visualizza il catalogo della directory CASA (che è subordinata alla directory radice).

*CDIR (*CD.) Crea una directory subordinata con il nome specificato. La directory creata è gerarchicamente 'figlia' della directory attuale. Per esempio:

*CDIR MARIA	crea una directory di nome MARIA e la subordina alla directory attuale.

*CLOSE (*CL.) Chiude tutti i file su disco attualmente aperti.

*COMPACT (*COM.) Riorganizza i file nella gerarchia delle directory in modo tale che gli spazi inutilizzati risultanti dalla cancellazione di file siano raggruppati in blocchi più vasti.

*COPY Copia un file da una directory all'altra. Per esempio:

*COPY $.TESTO $.BACKUP.VIEW

copia il file TESTO, presente nella directory radice, nella directory $.BACKUP.VIEW, cioè crea un file il cui nome di percorso è $.BACKUP.VIEW.TESTO.

*CREATE □ Riserva spazio per un file.

*DELETE (*DE.) Cancella il nome del file specificato. Per esempio:

*DELETE IDEA	cancella il file IDEA dalla directory attuale;
*DELETE $.CASA.B1	cancella il file B1 dalla directory CASA (anch'essa figlia della directory radice).

Una directory può essere cancellata solo se vuota.

*DESTROY (*DES.) Cancella un gruppo di file (usando eventualmente caratteri wildcard).

*DIR Seleziona la directory corrente.

*DISMOUNT (*DISM) Chiude tutti i file aperti prima di cambiare un disco.

*DRIVE (*DR.) Seleziona ed eventualmente attiva il drive specificato facendolo diventare drive corrente. Questo comando non dovrebbe essere utilizzato dai programmatori.

*DUMP (*DU.) □ Visualizza una stampa dump esadecimale del file specificato.

*EX	□	Visualizza le informazioni relative a tutti i file contenuti nella directory specificata.
*EXEC (*E.)		Fa sì che l'input al PC 128 S provenga dal file specificato piuttosto che dalla tastiera.
*FORMAT (*FO.)		Formatta il disco nel drive specificato usando la dimensione indicata.
*FREE (*FR.)		Visualizza la quantità di spazio libero su disco.
*INFO (*I.)	□	Visualizza informazioni relative ad un singolo file (o un gruppo di file) usando l'opzione dei caratteri wildcard.
*LCAT (*LC.)		Visualizza un catalogo della directory di libreria.
*LEX	□	Visualizza informazioni sui file contenuti in una directory di libreria.
*LIB		Seleziona la libreria sul drive e la directory specificata. Per esempio: *LIB $.UTILITA'
*LIST (*LI.)	□	Visualizza il disco specificato in formato GSREAD con numeri di riga.
*LOAD (*L.)	□	Carica in memoria il file specificato.
*MAP		Visualizza la mappatura dello spazio libero.
*MOUNT (*MOU.)		Inizializza un drive; solitamente usato quando si hanno più drive per passare dall'uno all'altro. *MOUNT 0 Inizializza il drive 0. Si noti che *MOUNT 0 è equivalente a *DIR :0.
*OPT1 (*O.1)		Seleziona il livello di colloquio durante le operazioni sui file.
*OPT4 (*O.4)		Imposta l'operazione dell'opzione di auto–caricamento.
*PRINT (*P.)		Visualizza una stampa dump in ASCII del file specificato.
*REMOVE (*RE.)		Equivale a *DELETE, ma nel caso in cui il file richiesto non si trovi non emette il messaggio di avvertimento "Not found".
*RENAME (*REN.)		Cambia il nome di un file su disco. Per esempio: *RENAME PROG1 PROG2

cambia il nome del file PROG1 in PROG2 (nella direc
corrente).

*RENAME può anche essere usato per spostare fis
mente (anziché farne una copia) un file da una direc
ad un'altra. Per esempio:

*RENAME $.BASIC.QUESTO $.SCARTI.QUELLO

sposta il file QUESTO dalla directory $.BASIC alla di
tory $.SCARTI e ne cambia il nome in QUELLO.

*RUN — Carica ed esegue un programma in linguaggio mac
na.

*SAVE (*S.) — □ Salva un blocco di memoria sul disco specificato.

*SPOOL (*SP.) — Fa sì che tutto l'output diretto allo schermo prima del
vio del comando venga ora dirottato sul file su disco s
cificato.

*SPOOLON — Fa sì che tutto l'output diretto allo schermo prima dell
vio del comando venga ora aggiunto al file su disco spe
ficato.

*SRLOAD — □ Carica il file specificato in un'area determinata della R
secondaria.

*SRSAVE — □ Salva un'area determinata della RAM secondaria sul
specificato.

*TITLE (*TIT.) — Imposta il titolo della directory corrente. Ad esempio:

*TITLE PROGRAMMA

Si noti che il titolo di una directory è cosa diversa
nome della directory stessa.

*TYPE (*TY.) — □ Visualizza nel formato GSREAD senza numeri di rig
contenuto del file su disco specificato.

*VERIFY (*V.) — Controlla che l'intera superficie del disco specificato
stata correttamente formattata e non presenti difetti.

Appendice 8

Parole chiave del BASIC

Per ogni parola chiave del BASIC è data di seguito una breve descrizione. Se è consentito l'uso di una forma abbreviata della parola chiave, tale forma è racchiusa in parentesi dopo la versione estesa della parola. Si noti che per certe parole chiave l'abbreviazione comprende una parentesi aperta: per esempio, LE. è equivalente a LEFT$(e non solo a LEFT$.

Molte delle parole chiave sono descritte dettagliatamente nel Capitolo 2. Le parole contrassegnate dal simbolo □ consentono l'uso di possibilità che vanno oltre l'intento di questo manuale e pertanto gli utenti che necessitano di un'informazione più completa sono invitati a consultare il manuale di riferimento.

ABS		Funzione che restituisce il valore positivo di un numero.
ACS		Funzione che restituisce l'arcocoseno, in radianti, di un numero fra –1 e 1 (compresi).
ADVAL (AD.)	□	Legge i dati dalla porta o dal buffer di joystick.
AND (A.)		Usato come operatore logico o a livello di bit.
ASC		Funzione che restituisce il codice numerico ASCII del primo carattere di una stringa.
ASN		Funzione che restituisce l'arcoseno, in radianti, di un numero fra –1 e 1 (compresi).
ATN		Funzione che restituisce il valore dell'arcotangente, in radianti, di un numero.
AUTO (AU.)		Comando per avviare la numerazione automatica delle righe.
BGET# (B.#)	□	Dà il codice del prossimo carattere di un file.
BPUT# (BP.#)	□	Registra su file il codice di un carattere.
CALL (CA.)		Mette in esecuzione una routine in linguaggio macchina.
CHAIN (CH.)		Carica e mette in esecuzione un programma BASIC.
CHR$		Funzione che restituisce il carattere corrispondente al valore ASCII specificato.
CLEAR (CL.)		Cancella tutte le variabili di programma in memoria, ad eccezione delle variabili intere residenti.
CLG		Riempie la finestra grafica con il colore attualmente usato per lo sfondo.
CLOSE# (CLO.#)		Chiude un file aperto.

CLS	Riempie la finestra di testo con il colore attualmente usato per lo sfondo.
COLOR o COLOUR (C.)	Imposta i colori per il testo e lo sfondo
COS	Funzione che restituisce il coseno di un angolo espresso in radianti.
COUNT	Variabile contenente il numero di caratteri stampati dall'ultimo ritorno a capo.
DATA (D.)	Usato insieme a READ, specifica un gruppo di dati utilizzati all'interno del programma.
DEF	Definisce una funzione o una procedura.
DEG	Funzione che converte la misura di un angolo da radianti in gradi.
DELETE (DEL.)	Cancella righe di programma.
DIM	Riserva spazio in memoria per un vettore o una matrice di dimensione data. DIV Effettua la divisione intera, trascurando il resto.
DRAW (DR.)	Traccia una linea a partire dall'ultimo punto grafico specificato fino a un nuovo punto.
EDIT	Richiama l'editor di testo opzionale.
ELSE	Parte dell'istruzione IF...THEN usata quando è richiesta una decisione alternativa.
END	Dopo questa istruzione il computer termina l'esecuzione. Il suo uso è opzionale quando questa istruzione è fisicamente l'ultima istruzione del programma.
ENDPROC (E.)	Indica la fine della definizione di una procedura.
ENVELOPE (ENV.)	□ Definisce un inviluppo sonoro.
EOF#	Funzione che riporta la situazione di raggiungimento di fine file.
EOR	Operatore logico di OR esclusivo.
ERL	Funzione che restituisce il numero di riga in cui si è verificato l'ultimo errore di programma.
ERR	Funzione che restituisce il codice dell'ultimo errore verificatosi.
ERROR (ERR.)	Parte dell'istruzione ON...ERROR.

EVAL (EV.)	Funzione che valuta una stringa come se fosse un'espressione BASIC.
EXP	Funzione che calcola *e* (cioè 2.7183...) elevato alla potenza specificata.
EXT#	Funzione che controlla la lunghezza di un file aperto.
FALSE (FA.)	Funzione che restituisce il valore 0. Usata nelle espressioni logiche.
FN	Usata nella definizione o nell'attivazione di una funzione.
FOR (F.)	Inizio del ciclo FOR...NEXT per eseguire iterativamente le istruzioni racchiuse fra FOR e NEXT.
GCOL (GC.)	Imposta il colore grafico che sarà usato dai successivi comandi grafici, e determina in che modo il colore interagisce con il colore di un qualsiasi altro punto situato nella stessa posizione dello schermo.
GET	Attende la pressione di un tasto e produce il codice ASCII corrispondente al tasto premuto.
GET$	Attende la pressione di un tasto e produce il carattere corrispondente al tasto premuto.
GOSUB (GOS.)	Esegue un sottoprogramma; una volta terminata la sua esecuzione, il sottoprogramma restituisce il controllo all'istruzione successiva a GOSUB. GOSUB è il predecessore meno potente di DEFPROC, e non consente il passaggio di parametri. L'istruzione è inclusa per compatibilità con il linguaggio BASIC di altri calcolatori.
GOTO (G.)	Trasferisce il controllo alla riga specificata.
HIMEM (H.)	□ Variabile usata per indicare la locazione di memoria libera utilizzabile dal programma avente indirizzo più alto. HIMEM può essere inizializzata dall'utente allo scopo di proteggere la memoria situata agli indirizi più alti, dove sono stati memorizzati i dati.
IF	Parte dell'istruzione IF...THEN. Il computer esegue le istruzioni che seguono la clausola THEN solo se la condizione che segue IF è vera.
INKEY	□ Attende la pressione di un tasto per un tempo determinato, e produce il codice ASCII relativo al tasto premuto. Il tempo è espresso in centesimi di secondo.

INKEY$ (INK.) □ Attende la pressione di un tasto per un tempo determinato, e produce il carattere corrispondente al tasto premuto. Il tempo è espresso in centesimi di secondo.

INPUT (I.) Attende uno o più dati dalla tastiera; il tasto RETURN deve chiudere l'introduzione.

INPUT LINE Accetta un'introduzione alfanumerica da tastiera contenente spazi o virgole anche in posizione iniziale e finale. Il tasto RETURN deve chiudere l'introduzione.

INPUT# (I.#) Legge i dati da un file aperto e li memorizza nelle variabili specificate.

INSTR (INS.) Ricerca nella stringa specificata una sottostringa. Se la sottostringa è presente nella stringa, ritorna la posizione del carattere da cui inizia la sottostringa.

INT Funzione che, dato un numero decimale, ritorna il numero intero più prossimo minore del numero originale.

LEFT$((LE.) Estrae la parte sinistra di una stringa.

LEN Funzione che ritorna la lunghezza di una stringa.

LET Assegna un valore a una variabile. L'uso della parola LET è facoltativo nel BASIC BBC.

LIST (L.) Lista il programma corrente. LISTO realizza l'indentazione in modo da rendere il programma più leggibile. LIST IF lista solo le righe di programma che contengono una particolare sequenza di caratteri.

LN Funzione che ritorna il logaritmo naturale di un numero.

LOAD (LO.) Carica in memoria un programma BASIC.

LOCAL (LOC.) Dichiara locali alla procedura o alla funzione le variabili che seguono. In tal modo il valore che esse assumono internamente alla funzione o alla procedura non vanno ad intaccare il valore da queste eventualmente assunto in altre parti del programma. I parametri passati a una procedura sono considerati automaticamente locali.

LOG Funzione che ritorna il logaritmo in base 10 di un numero.

LOMEM (LOM.) □ Variabile usata per indicare la locazione di memoria libera di indirizzo più basso che può essere utilizzata per memorizzare il valore delle variabili usate dal programma. LOMEM può essere reimpostata dall'utente.

MID$((M.) Estrae da una stringa data una sottostringa.

MOD Operatore che dà il resto intero dopo una divisione.

MODE (MO.)	Cambia la modalità di visualizzazione. Non può essere usato all'interno di una procedura o di una funzione.
MOVE	Sposta senza visualizzarlo il cursore grafico a una data posizione.
NEW	Cancella dalla memoria il programma attualmente residente che può essere recuperato tramite OLD.
NEXT (N.)	Parte di un ciclo FOR...NEXT; indica la fine delle istruzioni facenti parte del corpo del ciclo.
NOT	Operatore logico o a livello di bit.
OFF	Parte di una istruzione ON ERROR OFF; disabilita l'opzione di gestione dell'errore e fa sì che al verificarsi di un errore il computer emetta la sua segnalazione standard di errore e interrompe il programma.
OLD (O.)	Recupera un programma dopo un comando NEW o dopo la pressione di BREAK. Se in quest'ultimo caso viene emesso un messaggio 'Bad Program', il programma non è più recuperabile e deve essere ricaricato dal disco.
ON ERROR	Istruzione usata per pilotare il programma dopo che in esso il computer abbia rilevato un errore.
ON...GOTO oppure ON...GOSUB	Il valore della variabile che segue la parola chiave ON determina il comportamento del programma. Se il suo valore è 1, il computer passa ad eseguire il primo numero di riga della lista che segue l'istruzione GOTO/GOSUB; se il suo valore è 2, esegue l'istruzione al secondo posto in tale lista, e così via.
ON...PROC	Serve ad eseguire a scelta una delle procedure specificate.
OPENIN (OP.)	Apre un file per elaborarlo solo in input.
OPENOUT (OPENO.)	Apre un file per elaborarlo solo in output.
OPENUP	Apre un file rendendo possibile l'elaborazione sia in input che in output.
OR	Operatore logico o a livello di bit.
OSCLI	□ Passa una stringa al sistema operativo.
PAGE (PA.)	□ Variabile usata dal computer per indicare la locazione di memoria alla quale comincia il programma. PAGE può essere impostato dall'utente e, se usato con la dovuta attenzione, è possibile avere in memoria contemporaneamente più di un programma.

PI	Rappresenta il valore di p greco (3.141592653).
PLOT (PL.)	Esegue una funzione di tracciamento in funzione dei parametri che lo seguono (per la lista completa dei comandi di tracciamento grafico, vedere l'Appendice 9).
POINT((PO.)	Ritorna il valore corrispondente al colore di un punto grafico.
POS	Ritorna il valore della coordinata x alla quale si trova attualmente il cursore di testo.
PRINT (P.)	Invia caratteri allo schermo. Il formato della stampa è condizionato dall'uso dei simboli ; , , e ' ; inoltre, la stampa dei numeri è regolata dal valore della variabile intera @%.
PRINT# (P.#)	Invia a un file aperto i valori delle variabili specificate nella istruzione.
PROC	Definisce o richiama una procedura.
PTR#	Funzione che ritorna la posizione interna a un file nella quale avverrà la successiva operazione di lettura o scrittura di un carattere. L'utente può cambiare il valore di PTR# e può quindi leggere o scrivere ovunque nel file, realizzando in tal modo un accesso casuale ai record.
RAD	Funzione che converte la misura di un angolo da gradi a radianti.
READ	Legge i dati da un'istruzione DATA.
REM	Commento. Serve a documentare il programma. Le istruzioni REM sono ignorate dal computer durante l'esecuzione del programma.
RENUMBER (REN.)	Rinumera le righe di un programma BASIC assegnando una numerazione standard oppure quella specificata.
REPEAT (REP.)	Prima parte del ciclo REPEAT...UNTIL; questo ciclo esegue tutte le istruzioni situate fra REPEAT e UNTIL ed esce dal ciclo quando una o più condizioni sono soddisfatte. Un ciclo di questo tipo è sempre eseguito almeno una volta, anche se le condizioni di uscita sono verificate subito, poichè il controllo per l'uscita dal ciclo è effettuato alla fine dello stesso.
REPORT (REPO.)	Stampa il messaggio relativo all'ultimo errore verificatosi.
RESTORE (RES.)	Legge i dati a partire dal numero di riga successivo a questa istruzione.

RETURN (R.)	Indica la fine di un sottoprogramma richiamato per mezzo di GOSUB. Il computer torna ad eseguire nel programma l'istruzione seguente alla chiamata GOSUB che ha richiesto l'esecuzione del sottoprogramma.
RIGHT$((RI.)	Estrae la parte destra di una stringa.
RND	Funzione che genera un numero casuale. RND(1) genera un numero decimale fra 0 e 0.99999. RND(N) genera un numero intero casuale fra 1 e N (compresi).
RUN	Avvia l'esecuzione del programma residente in memoria.
SAVE (SA.)	Salva su disco il programma in memoria centrale.
SGN	Funzione che ritorna il segno del numero specificato, e cioè –1 se il numero è negativo, 0 se è 0 e +1 se il numero è positivo.
SIN	Funzione che ritorna il seno di un angolo espresso in radianti.
SOUND (SO.)	Produce un suono attraverso l'altoparlante interno.
SPC	Funzione usata solo in istruzioni PRINT o INPUT per produrre una serie di spazi.
SQR	Funzione che calcola la radice quadrata di un numero.
STEP	Parte di una istruzione FOR...TO...STEP. Consente di usare incrementi diversi da 1.
STOP	Interrompe un programma causando un messaggio di errore STOP non gestibile da utente.
STR$	Converte un numero nella rappresentazione di tipo stringa ad esso equivalente.
STRING$((STRI.)	Produce delle copie di una stringa composta da un numero qualsiasi (massimo 255) di caratteri.
TAB	Funzione usata in istruzioni PRINT o INPUT per posizionare il cursore di testo su una data colonna sullo schermo.
TAN (T.)	Funzione che ritorna il valore della tangente trigonometrica di un angolo espresso in radianti.
THEN	Parte di una istruzione IF...THEN.
TIME (TI.)	Imposta o legge in centesimi di secondo il valore di uno dei clock interni.
TIME$	Questa pseudo–variabile produce il risultato "Fri, 31 Dec 1999.23:59:59". Il tentativo di assegnare valori a questa pseudo–variabile non avrà effetto.

TO

Parte di una istruzione FOR...TO...NEXT.

TOP

□ Variabile che contiene la prima locazione libera di memoria dopo la fine di un programma BASIC. TOP ha normalmente lo stesso valore di LOMEM; a differenza di questo, il suo valore non può essere impostato dall'utente.

TRACE (TR.)

Visualizza i numeri delle righe di programma in esecuzione. E' usato per rilevare gli errori. TRACE OFF disabilita la funzione, TRACE ON la abilita.

TRUE

Funzione che ritorna il valore –1; usata nelle espressioni logiche.

UNTIL (U.)

Ultima istruzione di un ciclo REPEAT... UNTIL che segnala la fine del ciclo. Le istruzioni situate fra REPEAT e UNTIL sono eseguite più volte fino al verificarsi di alcune condizioni.

USR

□ Funzione che richiama una routine scritta in linguaggio macchina che, come una funzione, ritorna un valore al programma chiamante.

VAL

Funzione che converte una stringa nella sua rappresentazione numerica equivalente. La stringa viene esaminata fino al primo carattere non–numerico; perciò, ad una stringa il cui primo carattere non è un numero, corrisponde un valore 0.

VDU (V.)

Comando per ottenere diversi effetti sullo schermo.

VPOS (VP.)

Ritorna la coordinata y alla quale si trova attualmente il cursore di testo.

WIDTH (W.)

Imposta la larghezza massima delle successive righe di output.

Appendice 9

Codici PLOT

Il formato di un'istruzione PLOT è il seguente:

PLOT *codice,x,y*

e il suo effetto è il tracciamento di una linea fino al punto (x,y) in un modo determinato dal valore del codice. Un risultato identico può essere raggiunto usando:

VDU 25,*codice,x;y;* (notare l'uso del punto e virgola).

I codici PLOT consentiti e i loro effetti si possono trovare (in gruppi di otto codici) nella Tabella 1. I codici all'interno di uno stesso gruppo si ottengono aggiungendo un valore di scostamento al primo codice del gruppo. I valori di scostamento sono i seguenti:

0 Spostamento relativo al punto precedente.
1 Tracciamento relativo al punto precedente nel colore di primo piano usato attualmente per la grafica.
2 Tracciamento relativo al punto precedente nel colore logicamente inverso.
3 Tracciamento relativo al punto precedente nel colore dello sfondo usato attualmente per la grafica.
4 Spostamento alla posizione assoluta.
5 Tracciamento alla posizione assoluta nel colore di primo piano usato attualmente per la grafica.
6 Tracciamento alla posizione assoluta nel colore logicamente inverso.
7 Tracciamento alla posizione assoluta nel colore dello sfondo attualmente usato per la grafica.

La colonna riportante come intestazione '*Punto Precedente*' contiene il numero di punti che devono essere 'toccati' prima dell'esecuzione dell'istruzione PLOT corrispondente. Per esempio, per disegnare un rettangolo, bisogna prima posizionarsi su un vertice (usando MOVE o DRAW); nell'istruione PLOT si specificheranno le coordinate del vertice diametralmente opposto.

Esempi di vari comandi PLOT si possono trovare nel Capitolo 3, mentre informazioni più approfondite possono essere reperite nel manuale di riferimento.

Tabella 1

Codice PLOT	Effetto	Punto precedente
0–7	Linea continua, estremità incluse	1
8–15	Linea continua, estremità finale omessa	1
16–3	Segmento a linea–punto–linea, estremità incluse, modello ripristinato	1
24–31	Segmento a linea–punto–linea, estremità finale omessa, modello ripristinato	1
32–39	Segmento a linea continuo, estremità iniziale omessa	1
40–47	Segmento a linea continua, omesse entrambe le estremità	1
48–55	Segmento a linea–punto–linea, estremità iniziale omessa, mantenimento del modello.	1
56–63	Segmento a linea–punto–linea, entrambe le estremità omesse, mantenimento del modello.	1
64–71	Tracciamento di un punto	
72–79	Linea continua da sinistra a destra con colore diverso dallo sfondo	
80–87	Riempimento di un triangolo	2
88–95	Linea continua verso destra con il colore dello sfondo	
96–103	Riempimento di un rettangolo	1
104–111	Linea continua da sinistra a destra nel colore del primo piano	
112–119	Riempimento di un parallelogrammo	2
120–127	Linea continua verso destra nel colore diverso da primo piano.	
128–135	Riempimento fino all'incontro con un punto di colore diverso dallo sfondo	
136–143	Riempimento fino all'incontro con un punto di colore uguale al primo piano	
144–151	Circonferenza	1
152–159	Cerchio pieno	1
160–167	Arco di circonferenza	2
168–175	Segmento di circonferenza	2
176–183	Settore circolare	2
184–191	Copia/spostamento di un rettangolo:	
	184 Spostamento relativo	2
	185 Spostamento relativo del rettangolo	2
	186/187 Copia relativa del rettangolo	2
	188 Spostamento assoluto	2
	189 Spostamento assoluto del rettangolo	2
	190/191 Copia assoluta del rettangolo	2
192–199	Ellisse	2
200–207	Ellisse piena	2
208–255	Riservati	

Appendice 10

Informazioni sul BAS128

BASIC BBC basato su disco – BAS128

Questa versione del BASIC BBC, fornita sul disco Welcome, permette all'utente di accedere a 64K di memoria per i programmi BASIC.

Per caricare il BAS128 digitare:

*MOUNT RETURN
*LIB LIBRARY RETURN
*BAS128 RETURN

Lo schermo verrà cancellato e quindi apparirà lo schermo iniziale **BASIC BBC** insieme al messaggio **Bytes free 64K** (64K byte liberi).

Viene visualizzato il prompt > e si possono introdurre i comandi esattamente come con il BASIC BBC basato su ROM.

I nuovi utenti del personal computer PC 128 S dovranno leggere la sezione riguardante il BASIC in questo manuale prima di utilizzare il programma. Se si conosce già il BASIC BBC e si desidera scrivere propri programmi, si dovrà tenere conto delle differenze tecniche esistenti tra le versioni basate su ROM e quelle basate su disco. Il punto più importante è che la versione basata su disco viene caricata nell'area RAM che generalmente è occupata dai programmi utente. L'area da 64K della RAM secondaria addizionale viene resa disponibile per i programmi utente.

Di seguito vengono dettagliate le differenze tra i due linguaggi BASIC:

1. Quando si utilizza il BAS128, i 64K di memoria addizionale sono come uno spazio continuo di indirizzamento oltre al normale spazio di 64K del computer. Quindi gli indirizzi da &000000 a &00FFFF si riferiscono alla memoria normale e gli indirizzi da &010000 a &01FFFF si riferiscono ai 64K byte di RAM secondaria.

L'indirizzo di default di PAGE è &10000.
L'indirizzo di default di HIMEM è &20000.

Analogamente gli indirizzamenti di default di LOMEM e TOP vengono realizzati come per PAGE e HIMEM.

2. Le istruzioni simili a:

DIM codice 100

assegnano un valore di 17 bit a "codice". Possono quindi essere utilizzati operatori indiretti per leggere e scrivere dati nell'area di memoria riservata dall'istruzione DIM, ad esempio:

```
codice=27128;
PRINT codice?3;
$(codice+27)="Giulio e Cristina"
```

Gli operatori indiretti indirizzeranno la memoria normale se il puntatore calcolato è nella gamma da &000000 a &00FFFF e indirizzeranno la RAM secondaria se il puntatore è nella gamma da &010000 a &01FFFF.

3. L'assemblatore opererà utilizzando O% e P% come puntatori a 17–bit.

Se la destinazione per il codice macchina generata dall'assemblatore è nella gamma da &000000 a &00FFFF verrà fatto riferimento alla memoria normale. Gli indirizzamenti da &010000 a &01FFFF fanno riferimento alla RAM laterale.

Il contatore di programma viene regolato in modo che i riferimenti agli indirizzi da &010000 a &01FFFF indichino la RAM secondaria da &008000 a &00BFFF. I riferimenti da un banco di RAM secondaria ad un altro vengono rilevati, e viene indicato un errore "Bank", ad esempio:

```
P%=&13F00:[JSR &1CF34
```

provocherà un messaggio di errore "Bank". Durante la compilazione, qualsiasi codice generato sul confine di un banco viene rilevato, e viene indicato un errore "Wrap" dal momento che l'esecuzione non può passare direttamente da un banco al successivo.

4. Il modo Shadow verrà forzato (ignorando i MODI con valori bassi) in modo che BAS128 possa risiedere nella RAM tra &3000 e &8000. La memoria restante da OSHWM a &3000 può essere utilizzata da BAS128 come spazio di lavoro.

5. I comandi LOAD, SAVE e CHAIN utilizzano la routine OSGBPB.

Appendice 11

Comandi VIEW

Comandi dello schermo di comando

La maggior parte dei comandi possono essere abbreviati ai loro caratteri iniziali. Se consentita, l'abbreviazione minima è riportata fra parentesi dopo il nome del comando.

I comandi contrassegnati dal simbolo □ sono relativi a caratteristiche la cui completa descrizione esula dagli intenti del presente manuale. La loro descrizione completa può essere trovata nella guida utente di VIEW.

Notare che dallo schermo di comando VIEW è possibile lanciare comandi del sistema operativo e del sistema di archiviazione.

CHANGE (C)		Cerca tutte le ricorrenze di una data stringa e la sostituisce con un'altra. Per esempio: CHANGE/ACQUA/VINO
CLEAR (CL)		Toglie tutti gli indicatori dal testo.
COUNT (CO)		Conta il numero di parole presenti in memoria o comprese fra gli indicatori (se presenti).
EDIT (E)	□	Inizia l'editing di un file troppo grande per essere contenuto nella memoria centrale.
FINISH (F)	□	Termina una sessione di editing.
FIELD *n* (FI)	□	Assegna la funzione di tabulazione al tasto corrispondente al codice ASCII *n* (impostazioni di default = FIELD 9.)
FOLD	□	Abilita/disabilita la funzione che ignora la differenza fra maiuscole e minuscole nei comandi SEARCH, CHANGE e REPLACE. Se privo di parametri, segnala lo stato corrente.
FORMAT (FOR)		Formatta l'intero documento presente in memoria.
LOAD (L)		Carica il file specificato in memoria cancellando quello attuale. Per esempio: LOAD PROVA
MICROSPACE (MI)	□	Abilita la microspaziatura.

MODE (M) — Pone il computer nel modo di schermo specificato. Per esempio:

MODE 132

MORE (MO) □ Continua una sessione di editing.

NAME (N) — Attribuisce un nome al file presente in memoria, oppure modifica il nome del file. Per esempio:

NAME LUGLIO12

NEW — Cancella il testo dalla memoria.

PRINT (P) — Stampa il testo su modulo in continuo.

PRINT (P) — Se lanciato senza parametri stampa il testo attualmente in memoria; se seguito da uno o più nomi di file, ne stampa il contenuto.

PRINTER (PRINTE) □ Carica in memoria il driver della stampante specificata. Per esempio:

PRINTER LASER

QUIT □ Abbandona la sessione di editing.

READ (RE) — Legge un file in memoria fino alla fine del documento. Può essere usato per inserire il contenuto di un file nel documento in memoria a partire dal punto segnalato da un indicatore. Per esempio:

READ INDEX 1 — inserisce il contenuto del file INDEX nel documento corrente a partire dal punto segnalato dall'indicatore 1.

REPLACE (R) — Ricerca tutte le ricorrenze di una stringa e richiede all'operatore se intende sostituirla con un'altra. Per esempio:

REPLACE COMPUTER CALCOLATORE

SAVE (SA) — Salva il testo attualmente in memoria con il nome specificato. Per esempio:

SAVE MIO–CV — salva il file attualmente in memoria con il nome MIO–CV;
SAVE — salva il file in memoria con il nome già assegnato.

SCREEN (SC) — Visualizza sullo schermo il testo così come verrebbe stampato. Per esempio:

SCREEN LETTERA visualizza il file LETTERA;
SCREEN visualizza il file in memoria.

SEARCH (S) — Ricerca la stringa specificata. Per esempio:

SEARCH cane

CTRL + f1 (CONTINUA RICERCA) per continuare la ricerca.

SHEETS (SH) — Stampa il testo introducendo una pausa fra una pagina e l'altra per consentire all'utente di introdurre il nuovo foglio. Per esempio:

SHEETS LIBRI	stampa il file LIBRI;
SHEETS	stampa il file in memoria.

SETUP (SET) — Imposta una o più flag dello schermo di testo. Per esempio:

SETUP FI	seleziona la formattazione e l'inserimento, ma non la giustificazione.

WRITE (W) — Scrive un testo su disco usando il nome specificato. Questo comando è più lento di SAVE ma può essere usato con gli indicatori; per esempio:

WRITE PARTE 1 2	salva la parte di documento compresa fra gli indicatori 1 e 2.

Comandi memorizzabili

Questi comandi sono usati nello schermo di testo e sono posti nel margine di comando premendo SHIFT + f8 (EDITING COMANDO).

CE *testo*		Centra il *testo* fra i margini destro e sinistro.
RJ *testo*		Giustifica a destra il *testo*, cioè lo allinea al margine destro.
LJ *testo*		Giustifica a sinistra il *testo*, cioè lo allinea al margine sinistro.
DH	□	Definisce il capo–pagina.
DF	□	Definisce il piede–pagina.
HE ON/OFF	□	Abilita/disabilita la stampa del capo–pagina.
FO ON/OFF	□	Abilita/disabilita la stampa del piede–pagina.
DM *m*	□	Delimita l'inizio della macro *m*.
EM	□	Delimita la fine della macro.
SR *l v*	□	Imposta il registro *l* con il valore *v*.
PB ON/OFF		Abilita/disabilita le interruzioni di pagina (valore di default = ON).

PL *n*		Imposta a *n* righe la lunghezza della pagina (valore di default = 66)
TM *n*		Imposta a *n* righe il margine superiore della pagina (valore di default = 4).
HM *n*		Imposta a *n* righe il margine per il capo–pagina (valore di default = 4).
FM *n*		Imposta a *n* righe il margine del piede–pagina (valore di default = 4).
BM *n*		Imposta a *n* righe il margine inferiore della pagina (valore di default = 4).
LM *n*		Imposta a *n* spazi il margine sinistro per la stampa (valore di default = 0).
LS *n*		Imposta a *n* il numero di righe per ogni interlinea fra le righe di testo.
TS ON/OFF	□	Abilita/disabilita la stampa fronte/retro.
PE		Salto pagina. PE *n* effettua il salto pagina se *n* è maggiore del numero di righe che ancora restano da stampare nella pagina.
OP	□	Pagina dispari; emette un salto pagina se il numero della pagina è pari, due se dispari.
EP	□	Pagina pari; emette un salto pagina se il numero della pagina è dispari, due se è pari.
HT –/* *n*	□	Imposta il carattere di evidenziazione su *n*.

Appendice 12

Comandi di ViewSheet

La maggior parte dei comandi può essere abbreviata alle prime lettere. Se esistente, l'abbreviazione minima viene fornita tra parentesi accanto al nome del comando.

I comandi contrassegnati □ si applicano a funzioni non descritte in questo manuale. per informazioni consultare il manuale per l'utente di ViewSheet.

Notare che i comandi del sistema operativo e del sistema di archiviazione possono essere emessi dallo schermo di comando di ViewSheet.

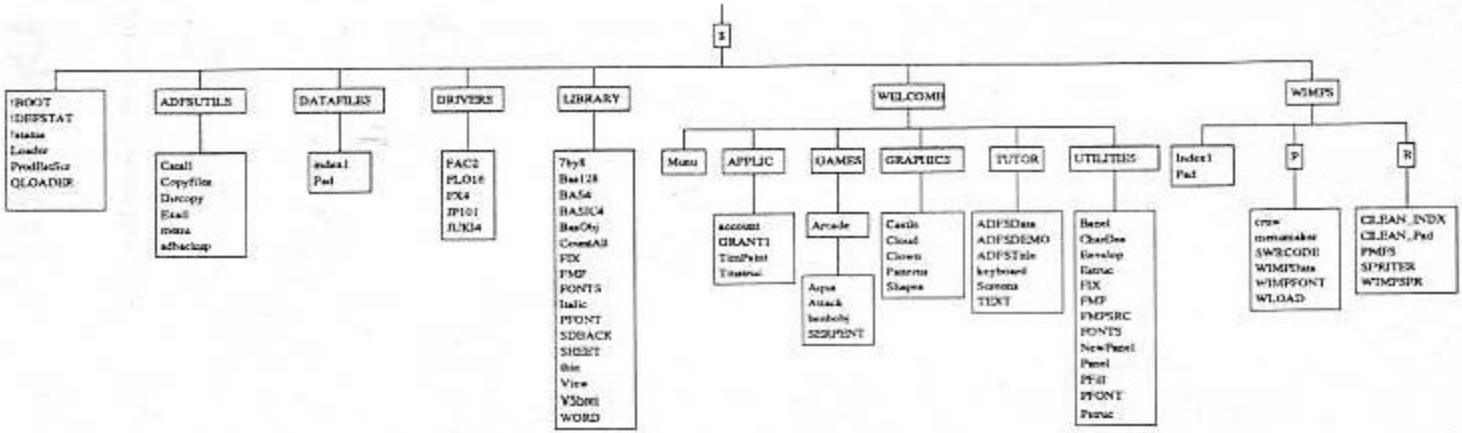
CREATE (CR)	□	Crea un file su disco che può essere letto (READ) o scritto (WRITE) all'interno del foglio.
HEADINGS (H)	□	Indica se sono impostati titoli definiti dall'utente.
HEADINGS OFF (H OFF)	□	Disattiva i titoli definiti dall'utente.
HEADINGS ON (H ON)	□	Attiva i titoli definiti dall'utente.
LOAD (L)		Carica in memoria il file specificato, sostituendo il precedente.
LW		Carica il file specificato o le definizioni di finestra.
MODE n (M)		Imposta il modo schermo specificato da n. Ad esempio: MODE 131
NAME (NA)		Assegna il nome specificato al foglio in memoria. Ad esempio: NAME SALARI2
NEW		Crea un nuovo foglio di lavoro.
PC		Stampa il contenuto di ogni cella occupata, con le coordinate.
PRINT (P)		Stampa il foglio in memoria.
PRINTER (PRINTE)		Carica in memoria il driver di stampante specificato. Ad esempio: PRINTER EPSON

PROTECT (PRO)	Indica se la protezione è attivata o disattivata.
PROTECT OFF (PRO OFF)	Disattiva la protezione.
PROTECT ON (PRO ON)	Attiva la protezione.
SAVE (SA)	Salva il foglio corrente con il nome specificato. Ad esempio: SAVE SOMME — salva il foglio corrente assegnandogli il nome SOMME; SAVE — salva il foglio corrente con il suo nome corrente.
SCREEN (SC)	Visualizza il foglio in memoria per mezzo delle finestre di stampa.
SW	Salva un file di definizioni di finestre.

Appendice 13

Albero del disco Welcome

Nella figura che segue è illustrata la gerarchia ad albero del disco Welcome versione 1.02 ISSUE 3.


$
!BOOT
!DEFSTAT
!status
Loader
ProdIntSrc
QLOADER
ADFSUTILS
Catall
Copyfiles
Discopy
Exall
menu
adbackup
DATAFILES
index1
Pad
DRIVERS
FAC2
PLO16
PX4
JP101
JUKI4
LIBRARY
7by8
Bas128
BA54
BASIC4
BasObj
CountAll
FIX
FMP
FONTS
Italic
PFONT
SDBACK
SHEET
thin
View
V5bmi
WORD
WELCOME
Menu
APPLIC
account
GRANT1
TimPaint
Tinstruc
GAMES
Arcade
Aqua
Attack
benbobj
SSRPENT
GRAPHICS
Castle
Cloud
Clown
Patterns
Shapes
TUTOR
ADFSData
ADFSDEMO
ADFSTile
keyboard
Screens
TEXT
UTILITIES
Banel
CharDes
Envelop
Extruc
FIX
FMP
FMPSRC
FONTS
NewPanel
Panel
PFII
PFONT
Pstruc
WIMPS
Index1
Pad
P
crow
mmumaker
SWRCODE
WIMPData
WIMPFONT
WLOAD
R
CLEAN_INDX
CLEAN_Pad
PMPS
SPRITER
WIMPSPR

Appendice 14

Caratteri accentati

à ■ seguito da :

é ■ seguito da {

è ■ seguito da [

ì ■ seguito da ¨

ò ■ seguito da j

ù ■ seguito da]

olivetti
PRODEST